STUDY OF SUB - BAND CODER FOR SPEECH SIGNALS

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

by
Capt S. K. DUBEY

to the

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
AUGUST 1984

8/8/82

CERTIFICATE

This is to certify that the thesis entitled 'STUDY OF SUB-BAND CODER FOR SPEECH SIGNALS' by Capt. S.K. Dubey has been carried out under our supervision and that it has not been submitted elsewhere for a degree.

(Mrs./S. Gupta)
Assistant Professor

(P.K. Chatterjee)
Professor

Department of Electrical Engineering
Indian Institute of Technology
KANPUR 208016, INDIA

25/8/84

ACKNOWLEDGEMENTS

I would like to express my deep sense of gratitude to my thesis supervisors, Dr. P.K. Chatterjee and Dr.(Mrs.) Sumana Gupta for having suggested this project to me and for all the invaluable guidance, constant encouragement and help given to me.

I am also grateful to Dr. V.P. Sinha for the valuable discussions I had with him on the design of filters and sampling rate converters.

My sincere and heartful thanks to Mr. P.G. Poonacha for teaching me many fundamentals of communication engineering.

I also wish to express my thanks to my friends Abhay Gupta, Neeraj Bhatnagar and Vineet Singh for their valuable suggestions in the software development.

My thanks are also due to Mr. J.S. Rawat for his excellent typing work.

Capt. S.K. DUBEY

ABSTRACT

Among the various techniques suggested for low bit-rate coding, sub-band coding seems to be very promising. It fills the gap between waveform coding techniques and vocoding techniques. It is also a simpler compromise to complex transform domain coders.

In sub-band coding, speech band is partitioned into 4 or 5 sub-bands and the sampling rate of each sub-band is decimated to the Nyquist sampling rate. Thus redundancy present in the speech is also reduced. The transmission bit-rate is kept low by implementing a coding technique such as APCM, which is most suited for encoding sub-band signals having poor sample-to-sample correlation.

This thesis addresses itself to the study of sub-band coders. Various guidelines for selection of parameters of sub-band coders have been discussed and suggestions made. A sub-band coder for speech signals has been simulated on the DEC system 1090 computer using FORTRAN language. Simulation studies have been done for 9.6 Kb/s and 16 Kb/s transmission rates. Conventional SNR's and segmental SNR's have been measured and the dynamic range of the coder has been assessed by varying the input level.

ADM has been implemented as an alternative to APCM coder to study whether simplicity and adaptability of ADM can be exploited for encoding sub-band signals.

CONTENTS

LIST OF FIGURES

Fig. No.	Caption	Page

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Speech is one of the main aspects in which the human race can be distinguished from the rest. The ability of speech in human beings inspired them to develop better means of communications.

With rapid developments in the field of science and technology, there is always an ever increasing demand on existing communication facilities. With shift from analog to digital communication, in the second half of twentith century, the communication engineers have to ensure the two following fundamental aspects in all their designs:

a) minimizing the number of bits which must be transmitted over the communication channel to convey the given information within given fidelity requirements

b) ensuring that bits transmitted over the channel are received correctly in the presence of interference of various types and origin.

The representation of the same amount of information with fewer bits, has twofold advantages. Firstly, the load on the existing communication channels can be reduced and secondly, the data storage capacity can be increased, at the

same time. This twofold objective can be achieved by low bit-rate coding. Low bit-rate coding requires exploitation of the fact that speech signals contain large amount of redundancy. This redundancy-removal or data-compression could be used to decrease bandwidth, to increase rate of transmission, to control probability of error and to reduce average signal power, besides low bit-rate generation.

In this thesis, a particular class of waveform coders called sub-band coder for low bit-rate speech transmission has been reported.

1.1 SPEECH DIGITIZATION:

Techniques of speech digitization are remarkably varied. Their difference depend mainly on the properties of speech and hearing, which are exploited in the design. These factors also influence the resulting bit-rate.

There is always a need for various speech coding techniques ranging from high to low bit-rates for various applications. Of course, the specific need will depend upon bandwidth and quality requirements. The range of bit-rates for transmission might be seen from Fig. 1.1. Speech digitization techniques can also be broadly classified from Fig. 1.1.

The left, high bit-rate side in Fig. 1.1, represents coding techniques that try merely to describe the acoustic waveform of the signal within given fidelity criterion. These techniques do no further coding of the source. This particular class of coders is called Waveform Coders. This class of coders includes high bit-rate pulse code modulation (PCM), adaptive pulse code modulation (APCM), linear delta modulation (LDM), adaptive delta modulation (ADM), differential pulse code modulation (DPCM) etc. These coders avoid the coder complexity and reproduce speech with a quality sufficient for commercial purpose. Important performance considerations in waveform coding are bit-rate, mean squared error (MSE) of reconstructed signal with respect to the original, dynamic range, implementation complexity and ruggedness towards transmission errors.

The right hand side of Fig. 1.1, represents the techniques which assume a speech generation model and transmit updates of the parameters of continuously changing model characteristics to achieve low bit-rates. These techniques include all analysis synthesis methods, the gamut of vocoders, linear and adaptive predictive coding, format synthesis and computer synthesis from stored text. The coders following these techniques are called parametric coders. Though this

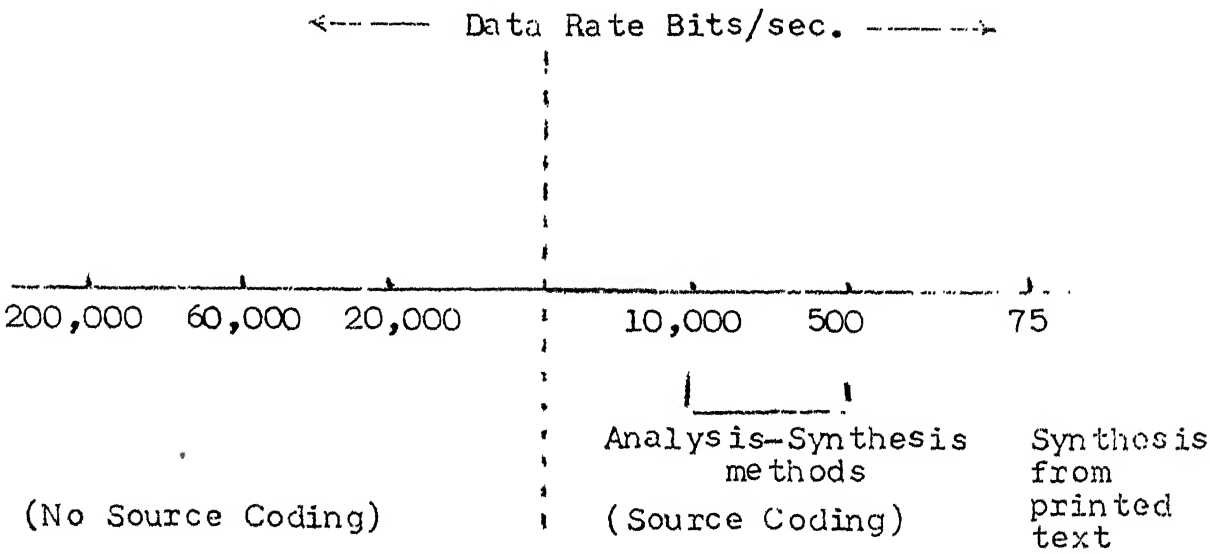


Figure 1.1: Spectrum of Bit Rates

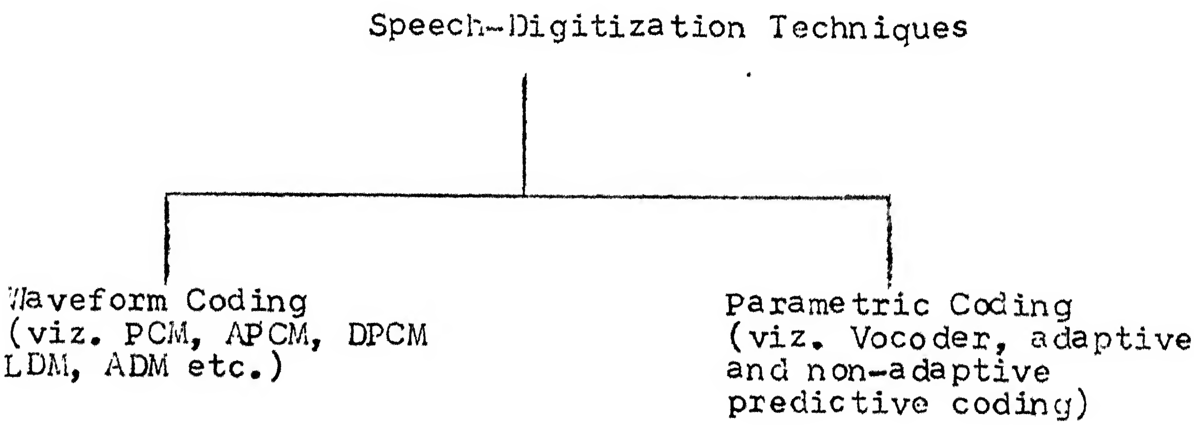


Figure 1.2: Speech Digitization Techniques

class of coders provideslow bit-rate but it is highly talker dependent, sensitive to background noise and expensive.

This classification has been illustrated in Fig. 1.2.

1.2 LOW BIT-RATE GENERATION:

The information capacity required to transmit or store the digital representation is given by

$$I = B.F_s = \text{Bit-rate in bits per second} \quad (1.1)$$

where,

F_s - Sampling rate (i.e. samples/second)

B - Number of bits per sample.

It is desired to maintain the bit rate as low as possible while maintaining a required level of quality. For a given speech bandwidth, the minimum sampling rate is fixed by sampling theorem. Therefore, the only way to reduce the bit-rate is to reduce number of bits per sample.

The various techniques discussed in Section 1.1 try to reduce the bit-rates in their own ways. Adaptive delta modulation (ADM) coders provide satisfactory performance at bit-rates above 32 Kb/sec and adaptive differential pulse code modulation (ADPCM) coders are suitable for 24-32 Kb/sec. range [1]. However, increasing need is being felt for efficient waveform coders in the 7.2 to 24 Kb/sec.range.

This is to benefit from the possibility of digital speech transmission over voice band provided by switched telephone lines and mobile VHF radios [2].

Though, the existing analysis and synthesis techniques provide low bit-rate but these are very expensive. Transform coding (TC) schemes also achieve the low bit-rate by suitable linear transformation of blocks of speech samples with the decorrelation being achieved in the transform domain [3]. But the transform coding schemes are also very complex.

A particular class of waveform coders called sub-band coder provides a compromise between sophisticated and expensive analysis synthesis techniques and high bit-rate waveform coding techniques such as ADM and adaptive differential PCM (ADPCM) in 7.2 to 16 Kb/sec. range [3,4,5]. Sub-band coder is also simpler compromise to complex transform domain coders.

1.3 SUB-BAND CODING:

Sub-band encoding uses a frequency domain analysis of input signal instead of time domain analysis. It is based on the partitioning of the speech band into sub-bands and encoding the sub-bands individually. This technique offers attractive possibilities for coding speech economically at

bit-rates in the range 7.2 to 16 Kb/sec. Combined with new emerging hard ware technologies such as CCD's (charged coupled devices), sub-band coding offers a practical and economical means of obtaining good quality in digital speech coding at low bit-rates. The partitioning of speech band into sub-bands enables to control and reduce quantizing noise in the coding. Each sub-band is quantized with an accuracy (bit-allocation) based upon perceptual criteria. As a result the quality of coded signal is improved over that obtained from a single full band coding of the total spectrum.

In sub-band coding, the speech band is partitioned into typically four or five bands by bandpass filters. Each sub-band is then lowpass-translated to dc, sampled at its Nyquist rate, and then digitally encoded using adaptive PCM (APCM) encoding. The encoded sub-band signals are multiplexed and transmitted. At the receiving end, the data is demultiplexed, decoded and the sampling rate is increased to the original sampling rate. The sub-band signals are again filtered by bandpass filters and then summed up to reconstruct the replica of the original signal.

Credit for early work on sub-band coder goes to Crochiere. Other early contributions came from Flanagan

and Webber [4,5,6,21].Crochiere, Flangan and Webber simulated the sub-band coder on computer and also carried out hardware implementation. They also compared the quality of the sub-band coder against that of ADPCM and ADM coders [6]. Goodman and Wilkinson suggested a modification in APCM algorithm, which reduced the transmission rate to 7.2 Kb/s [22]. Crochiere and Sambur further brought down the transmission rate of sub-band coder to 4.8 Kb/s by using variable band coding scheme [23].

1.3.1 Comparison of Sub-band Coding with other Waveform Coding Techniques:

Sub-band coding technique offers attractive possibilities for coding speech economically at bit-rates in the range of 7.2 to 16 Kb/sec. Informal listening tests were made by Crochiere, Webber and Flanagan [5] to compare the quality of the sub-band coder with that of full band encoding.

When 16 Kb/sec sub-band coder was compared with 16 Kb/sec ADPCM, listeners preferred sub-band coder in more than 94% cases. When the bit-rate of ADPCM coder was increased to 24 Kb/sec (3 bits/sample), the listeners rated sub-band encoded sentence as having higher quality in 34% cases. This clearly indicates that the quality of the 16 Kb/sec sub-band coder is clearly preferred over that of ADPCM at the same bit-rate.

When 9.6 Kb/sec. sub-band coder was compared with ADM coder at different bit-rates of 10.3, 12.9 and 17.2 Kb/sec., the listeners preferred sub-band coders in 96%, 82% and 61% cases respectively.

When bit-rates of ADPCM and ADM coders were increased to improve their quality then 16 Kb/sec. sub-band coder was found to have a quality comparable to 26 Kb/sec.ADPCM coder and 9.6 Kb/sec.sub-band coder had quality comparable to 19.2 Kb/sec.ADM coder.

Therefore, for same subjective quality, the sub-band coder has about a 10 Kb/sec. advantage over ADPCM and ADM coders.

1.4 ORGANISATION OF THE THESIS:

This thesis consists of six chapters. In Chapter 2, the sub-band coder-decoder has been discussed and the main simulation steps have been indicated. Chapter 3 deals with the various criterion for the selection of sub-bands and integer band sampling. In Chapter 4, various aspects of decimation and interpolation are presented. It also deals with the specifications of the lowpass filter used in the process of sampling rate conversion. Chapter 5 provides detailed discussions on APCM and ADM coders and decoders. In

Chapter 6, the simulation results have been given and conclusions are drawn on these results. It also includes suggestions for future work.

Under appendix A, input files have been discussed and signal flow-chart for the simulation program has been shown. Appendix B presents signal flow-chart for the filter design. Appendix C presents listing of the program.

CHAPTER 2

SUB-BAND CODER - SYSTEM DESCRIPTION

For digital transmission, a signal must be sampled and quantized. Quantization is a non-linear operation and produces distortion products that are typically broad in spectrum. Because of the characteristic of the speech spectrum, quantizing distortion is not equally detectable at all frequencies. Coding the signal in narrower sub-bands offers one possibility for controlling the distribution of quantizing noise across the signal spectrum. This also realizes an improvement in the signal quality.

In coding any waveform, there are two basic concerns: the sampling rate and the number of bits per sample. The product of these two quantities gives the bit-rate. To substantially reduce this bit-rate, without making any compromise in the quality, use must be made of the considerable redundancy present in speech in the form of non-uniform amplitude distribution, large correlation between successive samples and the consequent non flat spectrum.

The redundancy removal in sub-band coders is done in frequency domain by decimating the sampling rate of each sub-band to the minimum acceptable sampling rate i.e.

Nyquist sampling rate of the sub-band. The lowpass translated, Nyquist rated speech samples have very low sample-to-sample correlation, which minimizes the redundancy present in the speech samples.

The bit-rate in the sub-band coder is further reduced by using such encoding techniques which use the least number of bits per sample.

2.1 SUB-BAND CODING:

The simulation of the sub-band coder on DEC-10 computer involves the following steps:

i) Selection of sub-bands based on equal contribution to articulation index (AI) and to enable integer band sampling.

ii) Filtering the speech band into the sub-bands selected in step (i).

iii) Decimating the sampling rate of each sub-band to $2f_i$ where f_i is the bandwidth of ith sub-band.

iv) Implementing suitable encoding technique on the decimated samples in each sub-band. APCM and ADM coding techniques have been used.

v) Multiplexing the data of the sub-bands and inserting synchronization data.

vi) Providing synchronization detector and demultiplexing.

vii) Decoding the data of each sub-band.

viii) Interpolating the sampling rate of data in each sub-band to the original sampling rate.

ix) Filtering the outputs of the interpolators with another set of bandpass filters identical to those used in step (ii) in the coder side.

x) Summing of the filter outputs of all the sub-bands to give reconstructed replica $\hat{S}(n)$ of input speech samples $S(n)$.

Steps (v) and (vi) have been eliminated assuming that the encoded data is available as input to the decoder after undergoing multiplexing and demultiplexing with synchronization.

The block diagram of the sub-band encoder is presented in Fig. 2.1.

The functioning of the various sub-blocks of the sub-band coder and implementation of the above mentioned steps have been described in the following paragraphs.

An important element in the design of sub-band coder is the selection of sub-bands that permit integer

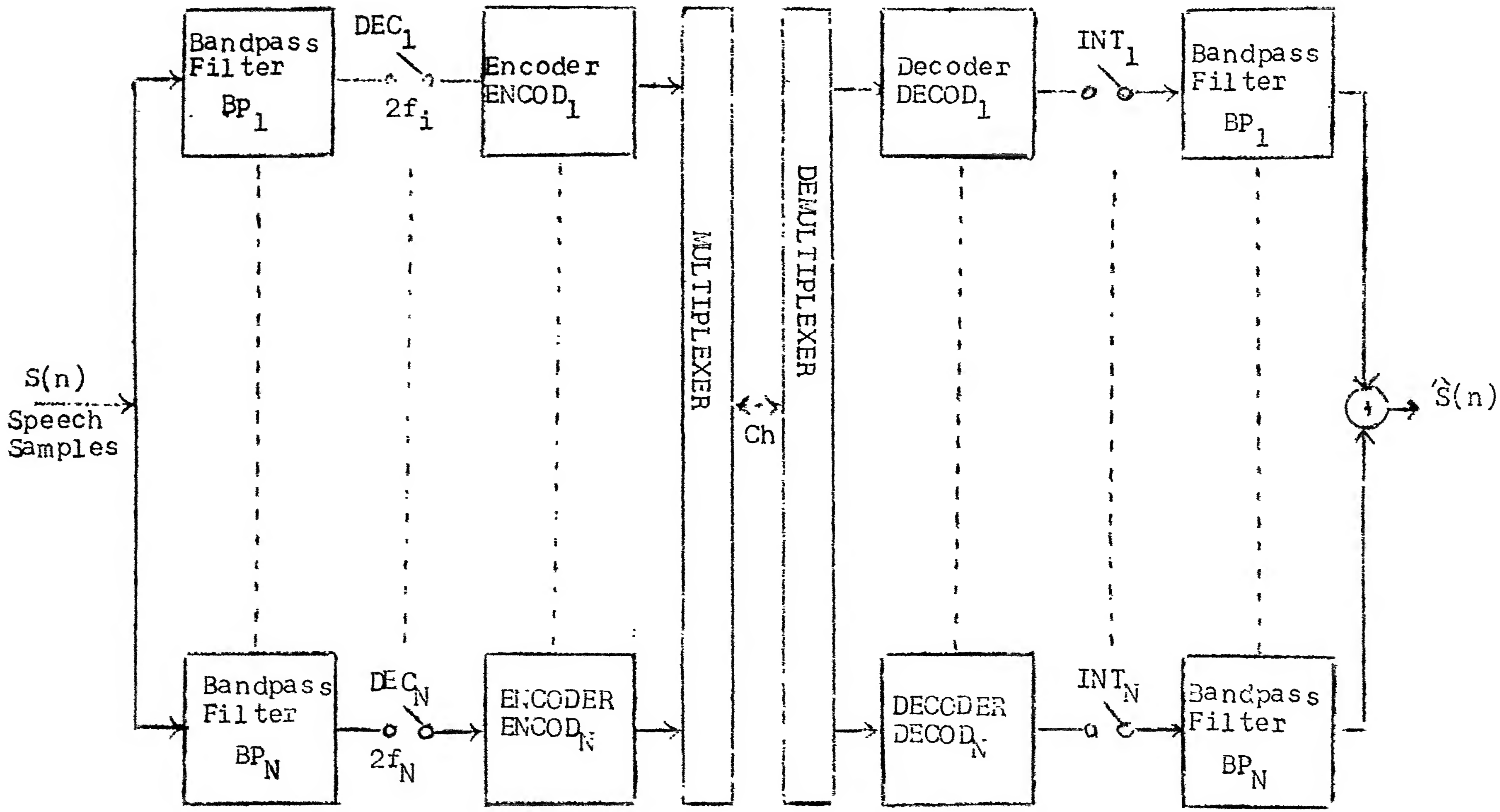


Figure 2.1: Sub-band Coder System Block Diagram.

band sampling and contribute equally to articulation index (AI). The speech band is partitioned into N sub-bands so selected by bandpass filters BP_1 to BP_N. The various aspects of selection of the sub-bands and integer band sampling are discussed in Chapter 3.

The output of each filter in the coder is decimated to a sampling rate $2f_i$ where f_i is the width of the sub-band and i refers to the ith sub-band. The decimators are denoted by DEC_1 to DEC_N in the figure. The decimated sub-band signals are digitally encoded by waveform encoders $ENCOD_1$ to $ENCOD_N$. The encoded samples of all the sub-bands are time-multiplexed for transmission over the digital channel.

At the receiver, the digital signals are demultiplexed and decoded by decoders, $DECOD_1$ to $DECOD_N$. The sampling rates of the decoder outputs are interpolated to the original sampling rate of S(n), the input speech signal by filling in with zero-valued samples. The interpolators are denoted by INT_1 to INT_N. The sub-band signals are reconstructed by filtering the outputs of the decoders with another set of bandpass filters identical to BP_1 to BP_N used in the coder. The outputs of these filters are then summed up to give a reconstructed replica $\hat{S}(n)$ of the original speech signal S(n).

2.1.1 Choice of Waveform Coders:

As discussed earlier, the sampling rate of sub-band signals is brought down to the lowest sampling rate, adequate to describe the information content within the band. The next step is to select such a encoding technique, which keeps up with the aim of keeping the transmission bit-rate low. Among the various waveform coding techniques available, adaptive pulse code modulation (APCM) has been found most suitable to accomplish the encoding of Nyquist rated sub-band signals having poor sample-to-sample correlation.

2.1.2 Input Signal:

The actual speech samples have been used as input. The original speech samples are sampled at 6 KHz and quantized to 11 bit accuracy. The bandwidth is 2940 Hz. The sentence used is 'Have you seen Bill'?

The digital speech samples are from Hewlett-Packard's FFT analyser of EE Department, Indian Institute of Science, Bangalore. First 2048 samples of the speech material have been used, which correspond to the segment 'Have you s....'. The segment contains a voiced section ('a') and an unvoiced section('s'). The waveform of speech segment is shown in Fig. 2.2. The sampling rate of these speech samples is

Fig. 2.1: Waveform of Speech Segment (2048 Samples)

increased to the sampling rate of the bandpass filters used in the sub-band coder.

2.2 ADVANTAGES OF SUB-BAND CODING:

The advantages of encoding in sub-bands over full band coding are listed as under.

1. By coding the speech signal in sub-bands, quantization noise can be contained in the sub-bands to prevent masking of one frequency range by quantizing noise in another frequency range.

2. By using seperate adaptation for each sub-band, the quantizer step size can be adjusted according to the energy level in each sub-band. Therefore, sub-bands with lower signal energy will have lower quantizer step size and contribute less quantization noise.

3. The bit-rate assigned to each sub-band can be optimized according to the perceptual importance of each individual band. In lower bands, where pitch and formant structure must be accurately preserved, a large number of bits per sample can be used for encoding, where as in upper bands where fricative and noise-like sounds occur in speech, fewer bits/ sample can be used.

CHAPTER 3

SELECTION OF SUB-BANDS AND INTEGER BAND SAMPLING

The selection of sub-bands involves a variety of considerations, such as the number of bands, bandwidth of the sub-bands and their location. The next step in processing the sub-bands is to perform lowpass and bandpass translation before coding. A variety of techniques exist for performing lowpass and bandpass translations. However, one approach is particularly attractive for hardware implementation since it eliminates the need of modulators. It is based on integer band sampling discussed in Section 3.2. The sub-bands have to be so selected as to contribute equally to the articulation index and at the same time permit integer band sampling. In this chapter, these issues have been discussed and sets of sub-bands for various sampling rates have been proposed.

3.1 CRITERIA FOR SELECTING SUB-BANDS:

A good compromise in the number of bands necessary for sub-band coding is generally found to be four or five bands. When less than four bands are used, bandwidths become too wide and do not allow for full utilization of the advantages of sub-band encoding. Designs with more than four or five bands tend to consume bandwidths in transition

bands of filter in addition to requiring more hardware for practical implementation.

A useful preliminary guideline for choosing sub-bands, is to partition the speech band into sub-bands that represent approximate equal contribution to the articulation index (AI) under noiseless conditions. In this way each sub-band contains a significant portion of important frequencies of the sub-band.

3.1.1 Concept of Articulation Index (AI):

Articulation index (AI) is defined as a weighed fraction representing for a given speech channel and noise condition, the effective proportion of the normal speech signal, which is available to a listener for conveying speech intelligibility.

Articulation Index is computed from acoustical measurements or estimates at the ear of a listener, of the speech spectrum and of effective masking spectrum of any noise which may be present. From the articulation index measurement tests a curve has been derived plotting articulation index (AI) versus frequency [7,8,9]. This curve is presented in Figure 3.1. This plot is used for selecting the sub-bands which contribute equally to the articulation index.

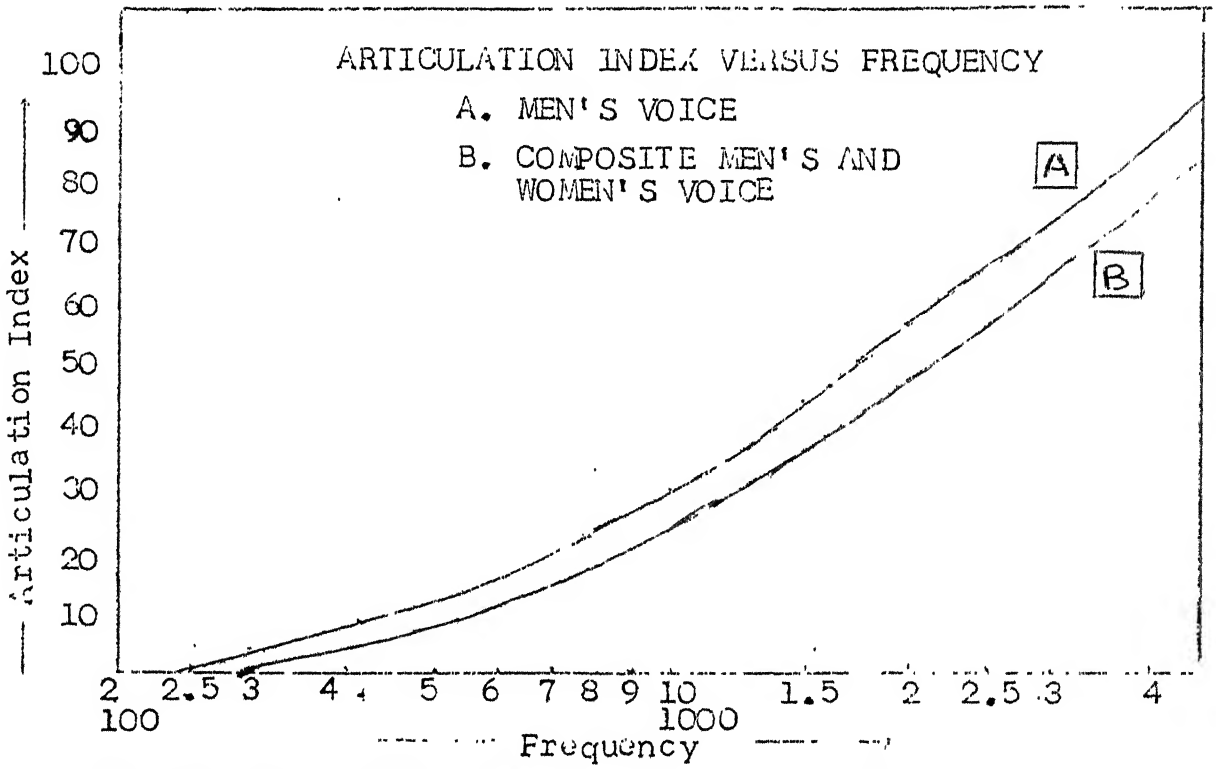


Figure 3.1: Articulation Index Versus Frequency

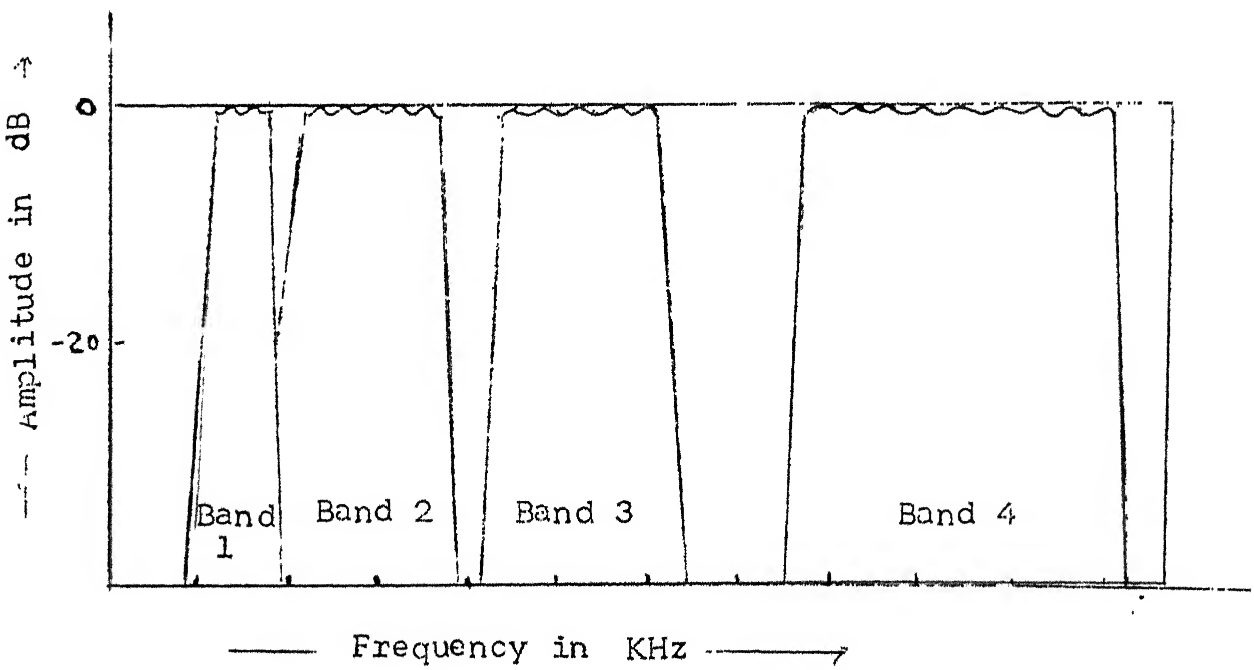


Figure 3.2: Partitioning of Speech Spectrum into Sub-bands

3.1.2 Location of Sub-bands:

Lower sub-bands should have narrower bandwidths and bandwidths should become progressively wider with increasing frequency. At lower bit rates, small gaps are permitted between bands to conserve bandwidth and bit rate, as shown in Fig. 3.2. If the gaps are too large, the noncontiguous bands produce a reverberant quality in the signal. However, some highly useful compromises can be achieved between transmission bit-rate and quality.

3.2 INTEGER BAND SAMPLING:

Sub-bands are lowpass translated before coding to facilitate sampling rate reduction and to realize any benefit which might accrue from coding the lowpass signal.

The lowpass translation can be performed in a variety of ways. In the present work a technique named integer band sampling has been used. This technique is better for hardware implementation and it also avoids the use of modulators as used in other conventional techniques for lowpass translation.

3.2.1 Sampling Theorem for Bandpass Signals:

Signals with bandpass spectra can also be represented by their sampled values. The spectrum of a bandpass signal is shown in Fig. 3.3.

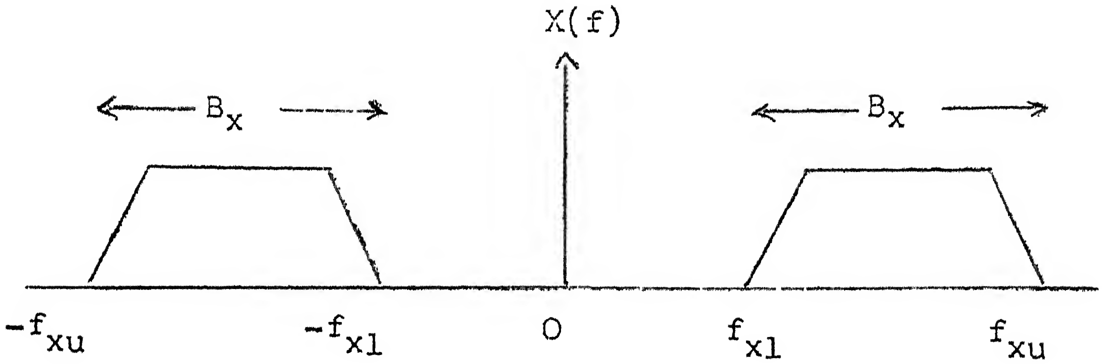


Figure 3.3: Spectrum of Bandpass Signal

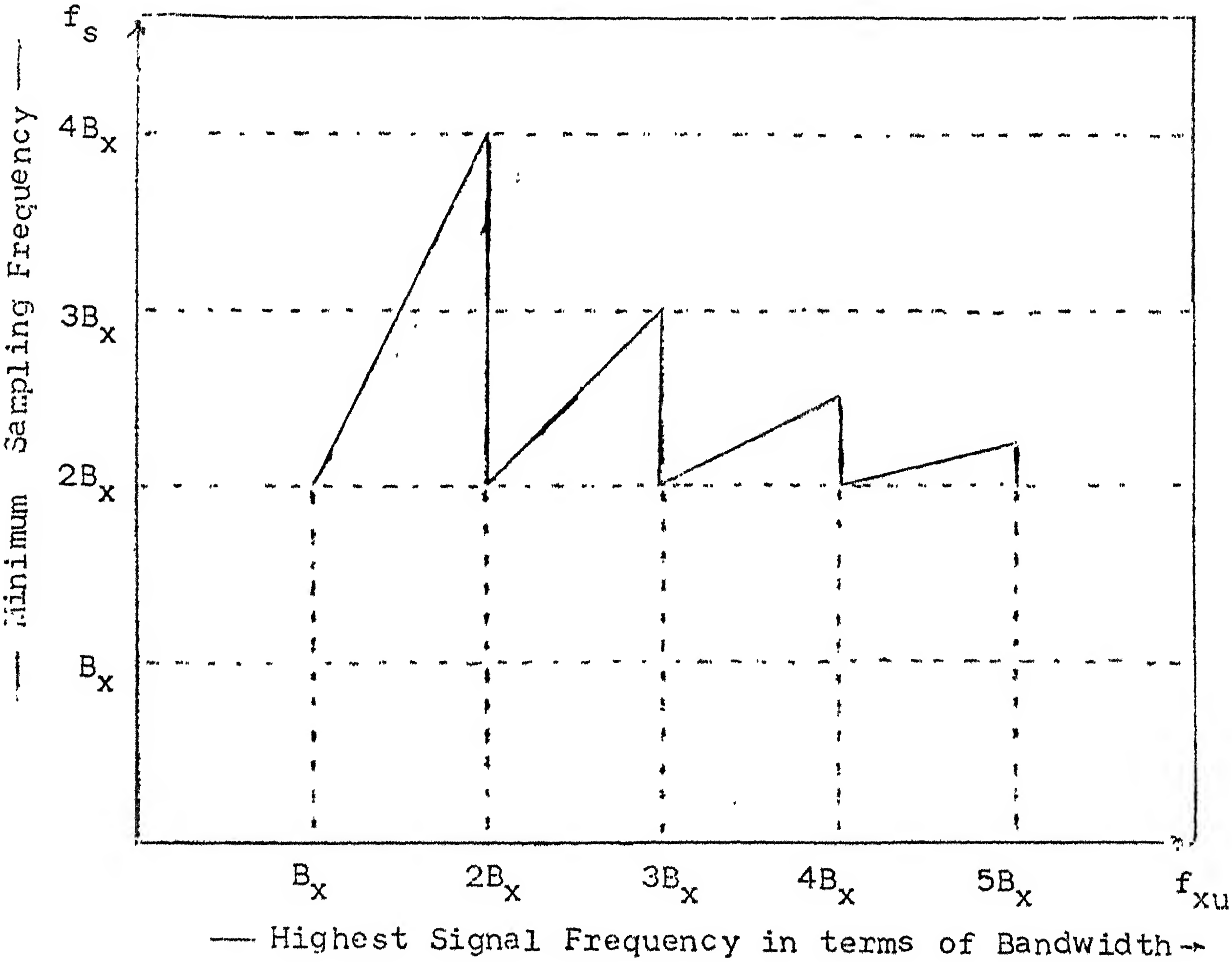


Figure 3.4: Minimum Sampling Frequency for a Signal Occupying a Bandwidth B_x.

If a bandpass signal $x(t)$ has a spectrum of bandwidth B_x and upper frequency limit f_{xu}, then $x(t)$ can be represented by instantaneous values $x(KT_s)$ if the sampling rate f_s is $\frac{2f_{xu}}{m}$, where m is the largest integer not exceeding f_{xu}/B_x. (Higher sampling rates are not always usable unless they exceed $2f_{xu}$). If the sample values are represented by impulses, then $x(t)$ can be exactly reproduced from its samples by an ideal bandpass filter $H(f)$ with the response

$$H(f) = \begin{cases} 1 & f_{x\ell} < |f| < f_{xu} \\ 0 & \text{elsewhere} \end{cases} \tag{3.1}$$

The sampling rate for a bandpass signal depends on the ratio f_{xu}/B_x. If $f_{xu}/B_x >> 1$, then the minimum sampling rate approaches $2\,B_x$. A sketch of f_{xu}/B_x versus f_s/B_x is shown in Fig. 3.4. The exact reconstruction occurs when

$$f_s = \frac{2\,f_{xu}}{m} \tag{3.2}$$

where m is an integer satisfying the equation

$$(f_{xu}/B_x) - 1 < m \leq f_{xu}/B_x \tag{3.3}$$

Integer band sampling technique is based on this theorem.

3.2.2 Integer Band Sampling Technique:

Integer band sampling technique is illustrated in Figure 3.5.

The signal sub-bands are chosen to have a lower cut-off frequency of mf_n and an upper cut-off frequency of $(m+1)f_n$, where m is an integer satisfying equation (3.3), and f_n is the bandwidth of the nth band. Typical values of m from 1 to 3 are most useful. The integer band sampling imposes the constraint that the ratio of upper to lower band edges of sub-bands be $(m_i+1)/m_i$ where m_i is an integer that may be different for different bands.

For hardware consideration, it is required that the sampling rates of sub-bands be derivable from a common clock. Furthermore, for digital or CCD hardware implementation, it is desirable to relate these sampling rates to the sampling rate of bandpass filters by ratios that are integers. Finally, the requirements for multiplixing digitally encoded sub-band signals dictate that the transmission bit-rates of each sub-band be a rational fraction of the total bit-rate so that the data can be framed and synchronized. Also a small fraction of this total bit-rate must be reserved for synchronizing and framing information.

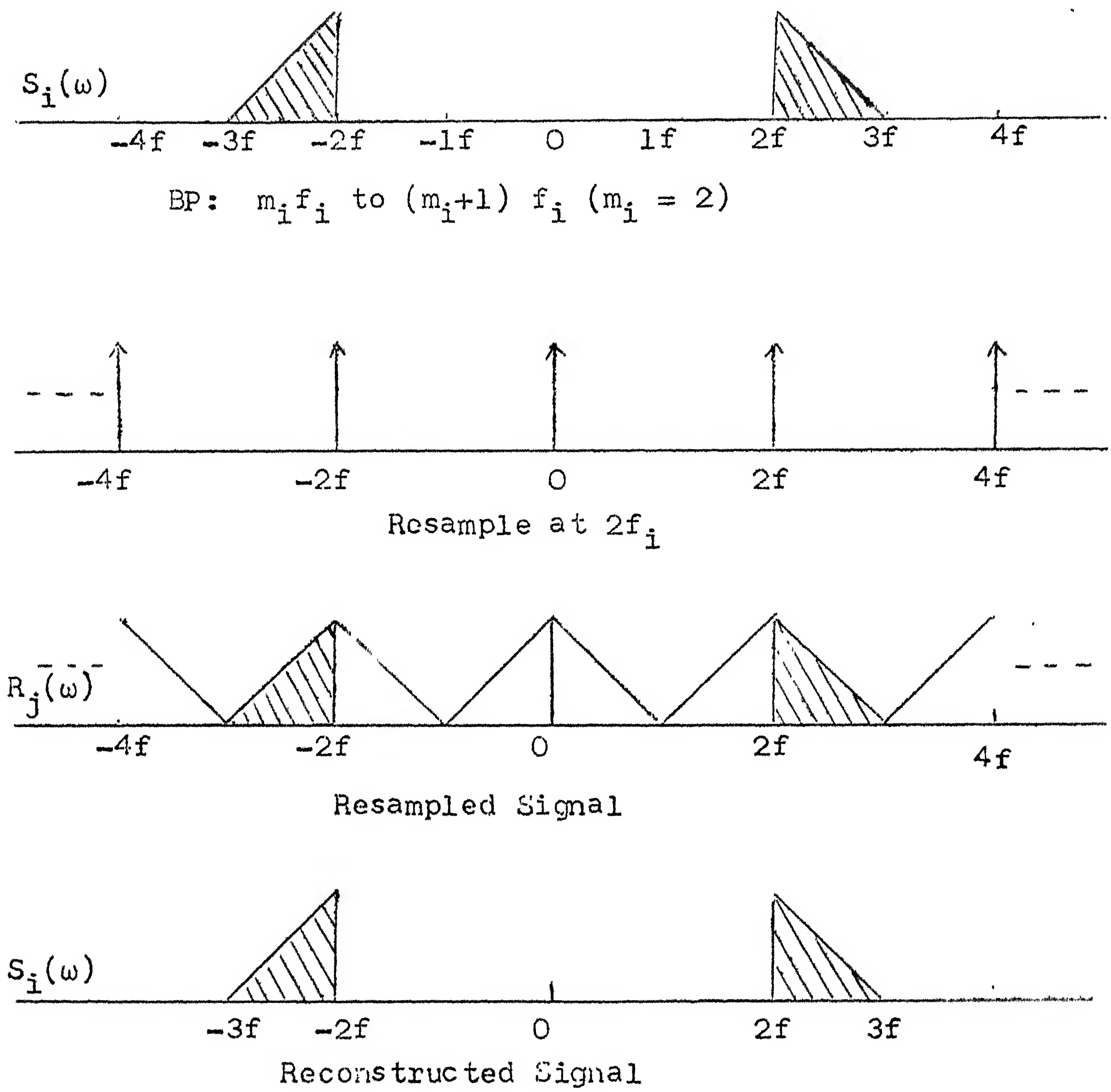


Figure 3.5: Integer-band Sampling Technique and a Frequency-domain Interpretation.

3.3 CHOICE OF SUB-BANDS FOR VARIOUS SAMPLING RATES:

The multitude of constraints discussed in the previous section greatly restricts the choice of sub-bands. To assist in the selection of sub-bands, it is useful to construct tables such as Table 3.1. It is assumed in this table that the sampling rate of the bandpass filter is 9.6 KHz. Column 1 indicates the integer decimation ratios that relate sub-band sampling rates to 9.6 KHz. Column 2 gives bandwidth, f_i, and column 3 gives 2 f_i sampling rates for possible sub-bands. Column 2 through 4 specify choices for band edges m_i f_i ($m_i = 1,2,3,...$). Therefore, all choices for sub-bands are discernible from the table once the sampling rate for the filter is chosen. It is to be ensured that the sub-bands so selected from such tables should contribute equally to the articulation index (AI) by mapping the sub-bands on the plot presented in Figure 3.1.

The final choice of sub-bands is further restricted by distribution of bits per sample across bands, the total transmission rate and the multiplexing requirements.

The transmission rate of each sub-band must be a rational fraction of the total bit rate so that the sub-band data can be multiplexed into a repetitive framed sequence. The lowest common denominator of these rational fractions,

Table 3.1

Choice of Sub-bands for Integer Band Sampling and 9.6 KHz Sampling Rate.

Decimation Ratio	f_i	$2f_i$	$3f_i$	$4f_i$
1	4800	9600	14400	19200
2	2400	4800	7200	9600
3	1600	3200	3800	6400
4	1200	2400	3600	4800
5	960	1920	2880	3840
6	800	1600	2400	3200
7	686	1371	2057	2743
8	600	1200	1800	2400
9	533	1067	1600	2133
10	480	960	1440	1920
11	436	873	1309	1745
12	400	800	1200	1600
13	369	738	1108	1477
14	343	686	1029	1371
15	320	640	960	1280
16	300	600	900	1200
17	282	565	847	1129
18	267	533	800	1067
19	253	505	758	1011
20	240	480	720	960
21	229	457	686	914
22	218	436	655	873
23	209	417	626	835
24	200	400	600	800
25	192	384	576	768

Table 3.2

Suggested Sub-bands for 9.6 and 7.2 Kb/sec. four Band Coder.

Sub-band Set No.	Sub-band Nos.	Sub-band Edges in Hz	
		From	To
I	1	229	457
	2	457	914
	3	1067	1600
	4	1920	2880
II	1	240	480
	2	480	960
	3	1067	1600
	4	1920	2880
III	1	253	505
	2	505	1011
	3	1067	1600
	4	1920	2880

including the fraction of transmission rate reserved for synchronization, determines the smallest frame size.

Table 3.2 presents a few choices of sub-bands that can be used for 9.6 and 7.2 Kb/s four band coder. The gaps appearing between the sub-bands give the coder a slightly reverberant quality.

Choice of sub-bands for 10.67 KHz sampling rate can be derived from Table 3.3. Table 3.4 presents a few possible sub-bands derived from Table 3.3 for 16 Kb/s five band coder.

3.4 DESIGN AND IMPLEMENTATION OF THE FILTERS:

The speech band is partitioned into sub-bands by bandpass filters. FIR filters of the order of 175-200 taps have been used. If wider transition regions are allowed, lower order filters can be used at the cost of an increased reverberant quality of the coder.

FIR filter is preferred over IIR filter due to their following properties:

i) exact linear phase

ii) no stability problem as encountered in IIR filters

iii) realization is efficient

iv) availability of efficient iterative design methods.

Table 3.3

Choice of Sub-bands for Integer Band Sampling and 10.67 KHz Sampling Rate.

Decimation Ratio	f_i	$2f_i$	$3f_i$	$4f_i$
1	5333	10667	16000	21333
2	2667	5333	8000	10667
3	1778	3556	5333	7111
4	1333	2667	4000	5333
5	1067	2133	3200	4267
6	889	1778	2667	3556
7	762	1524	2286	3048
8	667	1333	2000	2667
9	593	1185	1778	2370
10	533	1067	1600	2133
11	485	970	1455	1939
12	444	889	1333	1778
13	410	821	1231	1641
14	381	762	1143	1524
15	356	711	2133	1422
16	333	667	1000	1333
17	314	627	941	1255
18	296	593	889	1185
19	281	561	842	1123
20	267	533	800	1067
21	254	508	762	1016
22	242	485	727	970
23	232	464	696	928
24	222	444	667	889
25	213	427	640	853
26	205	410	615	821
27	198	395	593	790
28	190	381	571	762
29	184	368	552	736
30	178	356	533	711

Table 3.4

Suggested Sub-bands for 16 Kb/s Five Band Coder

Sub-band Set No.	Sub-band No.	Sub-band Edges in Hz	
		From	To
I	1	178	356
	2	296	593
	3	533	1067
	4	1067	2133
	5	2133	3200
II	1	190	381
	2	333	667
	3	593	1185
	4	1067	2133
	5	2133	3200

3.4.1 Computer Aided Design of Linear Phase FIR Filter:

FIR filters are characterized by their finite duration impulse response, $\{h_o, h_1, h_2, \ldots, h_K\}$. Their transfer function is a polynomial in z^{-1}.

$$H(z) = \sum_{k=o}^{K} h_k \, z^{-k} \tag{3.4}$$

To design a filter of this type, one selects the coefficients, $\{h_k\}$, so that the transfer function has a frequency response $H(e^{j\lambda})$, $-\pi \leq \lambda \leq \pi$, that approximates the design specifications within certain tolerances.

If the input signal is sampled at F_s, then the analog frequency f in Hz is related to the digital frequency through

$$\lambda = 2\pi f/F_s \tag{3.5}$$

The desired frequency response H(z) may be specified either in terms of real and imaginary parts, or equivalently in terms of its amplitude and phase, i.e.

$$H(e^{j\lambda}) = \sum_{k=o}^{K} h_k \, e^{j\lambda k} \tag{3.6}$$

It can be shown that, if the phase of $H(e^{j\lambda})$ is linear in λ, the impulse response must be symmetric in the sense that

$$h_k = h_{K-k} \tag{3.7}$$

In this case, equation (3.6) can be rewritten as

$$H\,(e^{j\lambda}) = [\sum_{n=o}^{N} a_n \cos(n\lambda)] \quad e^{-j\lambda K/2} \qquad (3.8)$$

for even K, where

$$N = K/2$$
$$a_o = h_N$$
$$a_n = 2\,h_{N-n}, \quad n = 1,2,3,\ldots,N \qquad (3.9)$$

For odd K, equation (3.6) can be written as

$$H(e^{j\lambda}) = [\sum_{n=o}^{N} a_n \cos\,(n-1)/2)\lambda]\ e^{-j\lambda K/2} \qquad (3.10)$$

where,

$$N = (K+1)/2\ ; \quad a_n = 2\,h_{N-n} \qquad (3.11)$$

Leaving aside for the moment the linear phase term $e^{j\lambda K/2}$ in equations (3.10) and (3.11), it is seen that the frequency response of the filter is given by a real cosine series, the coefficients of which are simply related to the impulse response. The linear phase delay is determined by the length of the impulse response only. The problem of the design of this type of filter becomes, therefore, one of the finding the values h_k so that the cosine series in eqn.(3.8)a equation (3.10) matches the desired function of λ as closely

as possible. This approach, due to McCellan et.al. [10] is very useful in computer aided design for a wide class of FIR filters, and the same has been used in this work for designing bandpass filters for the partitioning of speech band and as lowpass filters in sampling rate converters.

The parameters of the bandpass filters are shown in Figure 3.6. The sub-band covers the frequency range from $m_i f_i$ to $(m_i+1)\ f_i$. For practical reasons the filter passband must have a slightly narrower frequency range from $m_i f_i + \Delta f$ to $(m_i+1) f_i - \Delta f$. A transition region, Δf, of the order of 50 to 60 Hz was used in simulation with good results. A passband ripple of 0.173 dB and a filter stop-band attenuation of the order of 46 dB gave satisfactory results in simulation. Seperate bandpass filter has been used for each sub-band, instead of bank of filters for the partitioning of speech band into sub-bands. This avoids the implementation complexity of the bank of filters.

3.4.2 Implementation of FIR Filters:

For implementing FIR bandpass filter, a non-recurrsive realization has been used. In the non-recurrsive realization, the present filter output y_n is obtained explicitly in terms of only past and present inputs, i.e., previous

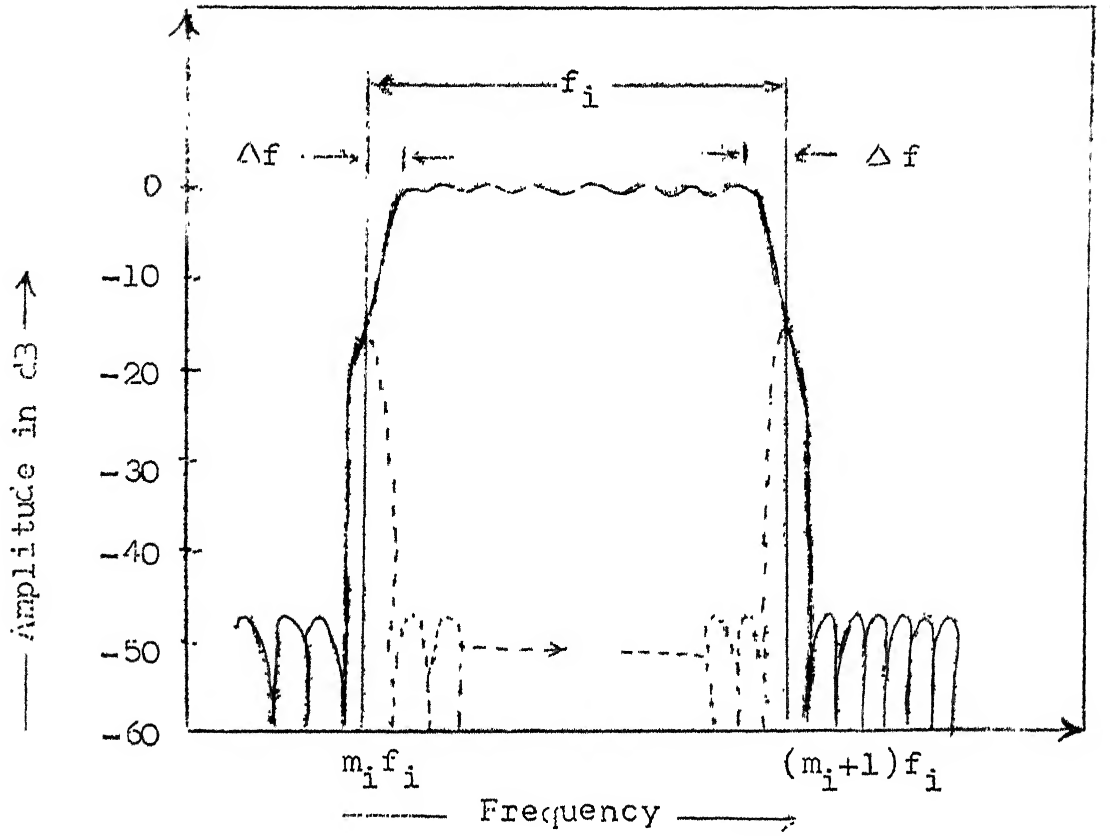


Figure 3.6: Parameters of the Bandpass Filters.

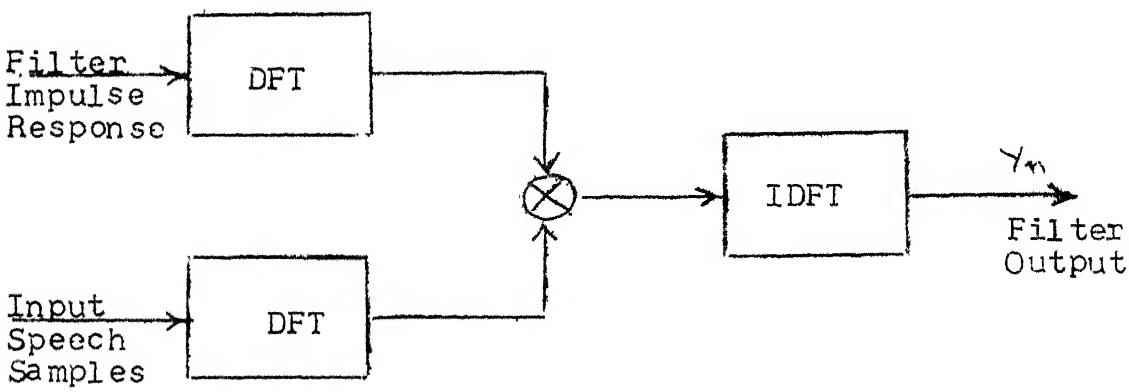


Figure 3.7: Filter Implementation.

outputs are not used to generate the present output. The representation of non-recurrsive realization can be written as

$$y_n = F(x_n, x_{n-1} \ldots\ldots) \qquad (3.12)$$

Fig. 3.7 presents the block diagram of filter realization. The discrete Fourier transforms (DFT) of input speech samples and filter impulse response were multiplied and the inverse discrete Fourier transform (IDFT) of the product gives the filter output. For taking DFT's and IDFT's, FFT subroutines of IMSL library have been used for faster execution.

CHAPTER 4

DECIMATION AND INTERPOLATION

The process of interpolation and decimation are fundamental operations in digital signal processing. The process of decreasing the sampling rate is known as decimation and increasing the sampling rate is referred to as interpolation in this context.

In sub-band coders, the sampling rate of each sub-band speech signals is brought down to the Nyquist sub-band sampling rate for low bit-rate transmission. This operation is performed by the decimation process. For reconstructing the replica of the original speech from the low bit-rate representation, the sampling rate of the decoded sub-band signals is again increased to the original sampling rate of the input speech samples. This operation is performed by the process of interpolation.

Two types of computational issues generally arise in the design and implementation of decimation and interpolation systems. The first issue involves the design of appropriate filter around which decimation or interpolation is based. The second issue involves the actual implementation of the decimation or interpolation processing.

The design of decimation or interpolation filter involves the use of lowpass digital filters. Such filters can be designed in a variety of ways, e.g. window designs, equiripple designs etc. In this thesis, the computer aided design for designing optimum FIR linear phase filter [10] has been used. This approach has been already mentioned in Section 3.4.

The implementation of decimators or interpolators involves the implementation of the digital filters in which the input and the output sampling rates are different. Because of this difference in sampling rates, a straight-forward implementation of FIR digital filter structure is not practical. Instead, special considerations must be taken in choosing the various specifications of digital filter and in efficiently implementing the digital filter.

A general module of sampling rate converter has been discussed in the following section. The various specifications for lowpass filter to fulfil the requirements of the process of decimation and interpolation are discussed in Section 4.2.

4.1 SAMPLING RATE CONVERTER:

Fig. 4.1 shows the sampling rate converter, which can be used to decimate as well as interpolate the sampling rates of sub-band signals.

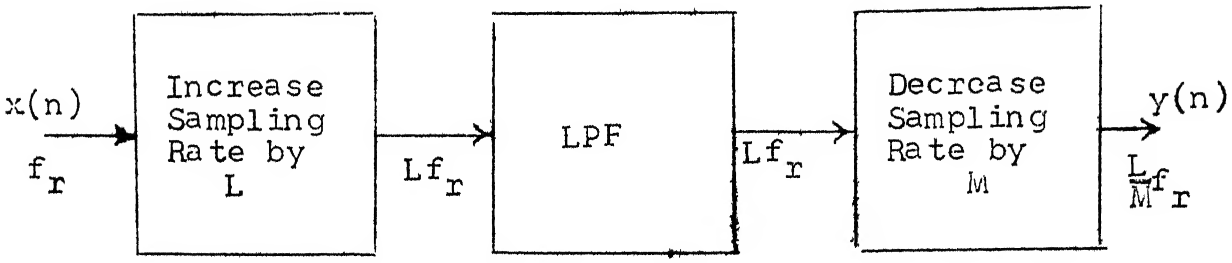


Figure 4.1: Sampling Rate Converter

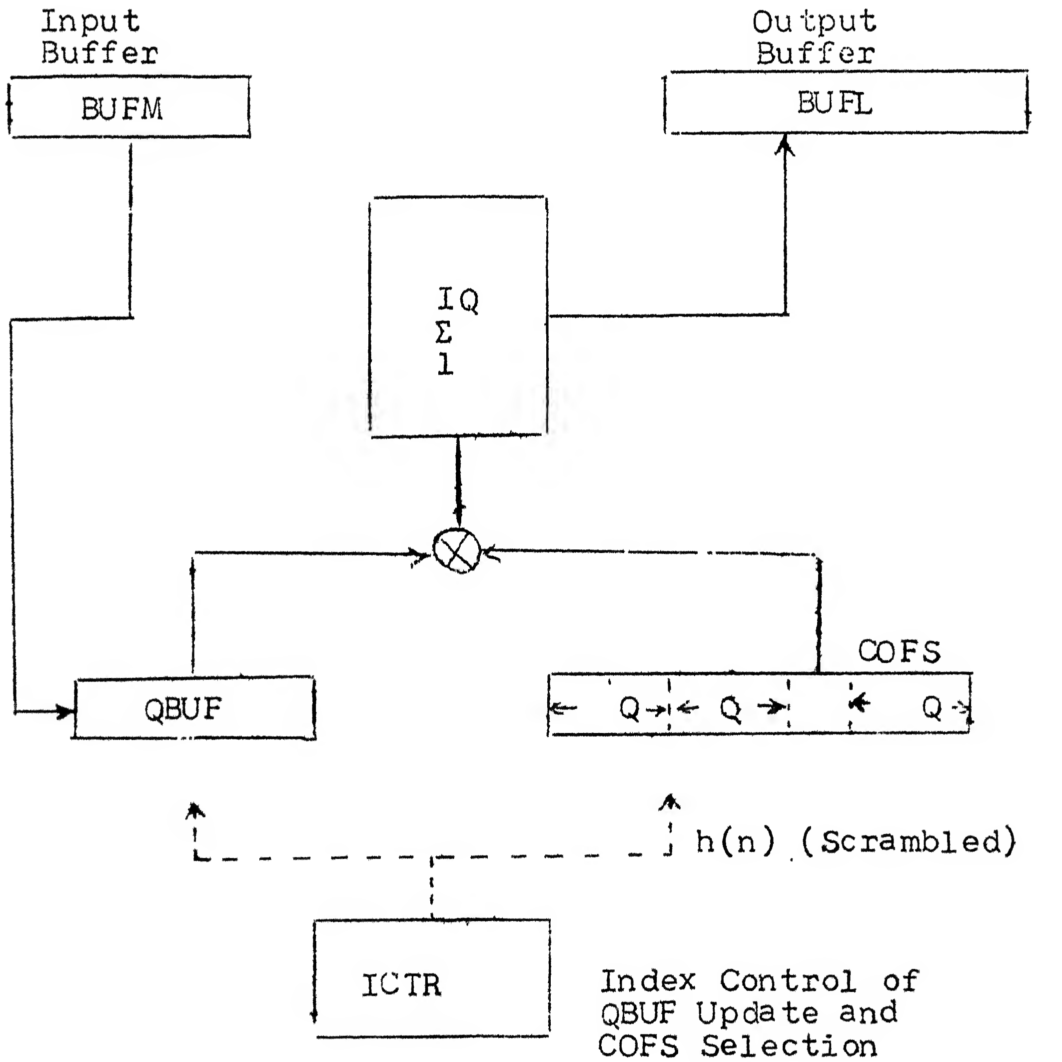


Figure 4.2: Flowchart of Program for Sampling Rate Conversion.

The sampling rate converter converts the sampling rate of sub-band signals by factors of L/M where L and M are arbitrary positive integers. The input sampling rate f_r is increased by a factor L by inserting L-1 zero valued samples between each pair of input samples. The insertion of zeroes is only conceptual and in practice, the input sequence $x(n)$ is treated appropriately, i.e., it is assumed that behind each $x(n)$ there are L-1 zero valued samples, when computing an output. This signal is then filtered with an FIR lowpass filter whose stopband cutoff frequency is $f_r/2$ or L $f_r/(2M)$, whichever is smaller. The output signal of this filter is then reduced by a factor of M by keeping only one out of every M samples, i.e., only the Mth sample is computed. By the elimination of computing unnecessary output samples it has been shown in [13], that the output y(n) can be computed by the relation

$$y(n) = \sum_{K=0}^{Q-1} h(KL + (nm) \oplus L)\, x\left(\left[\frac{nM}{L} \right] - K\right) \qquad (4.1)$$

where $(\quad) \oplus L$ implies the quantity in parenthesis modulo L and [ ] corresponds to the highest integer less than or equal to the number in brackets. The sequence h(n), n = 0,1,2....N-1 represents the coefficient of the FIR filter and N is the number of taps in the filter such that

$$N \leq QL \qquad (4.2)$$

and Q is an arbitrary positive integer.

Fig. 4.2 shows the flowchart of the program which implements the relation (4.1). Input data x(n) is supplied through BUFM and output data y(n) is received through BUFL. QBUF stores the necessary internal state variables and COFS stores the coefficients h(n). ICTR is a control memory which is generated by the initialization program and is used to control the indexing of data and coefficients in the program. The program is based upon the one given in reference [15]. The program performs decimation when L is equal to 1 and interpolation when M is equal to 1.

4.2 SPECIFICATIONS FOR LOWPASS FILTER:

The purpose of decimating the sampling rate of sub-band signals is to obtain a reduced sample rate, adequate to describe the information content only within the band of interest. To avoid aliasing at this reduced sampling rate, it is necessary to filter the original signal with a lowpass filter and then only sampling rate reduction could be achieved as described in Section 4.1.

Similarly for regeneration of replica of original speech samples, the sampling rate of sub-band signals are

increased to the original sampling rate. Once the sampling rate is increased then it is necessary to remove the images of the signal spectrum that are centred at integer multiples of $2\pi/T$, while leaving the frequency below $\frac{\pi}{T}$ unaltered.

Therefore, whether it is a process of decimation or interpolation, lowpass filtering is required to avoid aliasing. Since it is impossible to realize an ideal lowpass filter, an ideal filter must be approximated.

FIR filter is preferred in this case also as compared to IIR filters due to the following reasons:

1. An ideal interpolator/decimator has zero phase or almost linear phase.FIR filter can provide exactly linear phase. These filters are optimal in the sense that the width of transition band between passband and stopband is minimum for given values of passband and stopband ripple and specified passband and stopband cutoff frequencies. Thus with FIR filters, the interpolation error due to phase non-linearity can be zero and error due to amplitude distortion can be made arbitrarily small.

2. Though IIR filters have recurrsive realization which serves computation time, the particular

nature of sampling rate conversion problem makes FIR filters a better choice.

In the following sub-sections, the specifications of the lowpass filters are discussed.

4.2.1 Choice of Passband and Stopband Cutoff Frequencies:

The choice of passband and stopband depends upon the Nyquist frequency (Ω) of the incoming sub-band signal.

In passband i.e. $0 \leq \omega \leq \omega_p$, the frequency response should be close to 1.0. If sampling rate is increased by a factor of L, the filter gain must approximate L in passband, instead of 1.0. This is achieved by multiplying the impulse response samples of the filter by L [14].

An error of $\pm \delta_p$ is allowed in the passband and the frequency response is required to be within $\pm \delta_s$ of zero in the stopband, as shown in Fig. 4.3.

If the original sampling period is such that $\frac{\pi}{T} \simeq \Omega$ then ω_p must be close to $\frac{\pi}{T}$, as illustrated in Fig. 4.3. With the result, that the transition band $\omega_p \leq \omega \leq \omega_{s1}$ is bound to be very narrow. It implies that a large value for the length of the filter will be required. In such cases, it is reasonable to define only one stop band

$$\omega_{s1} \leq \omega \leq \frac{\pi}{T} \tag{4.3}$$

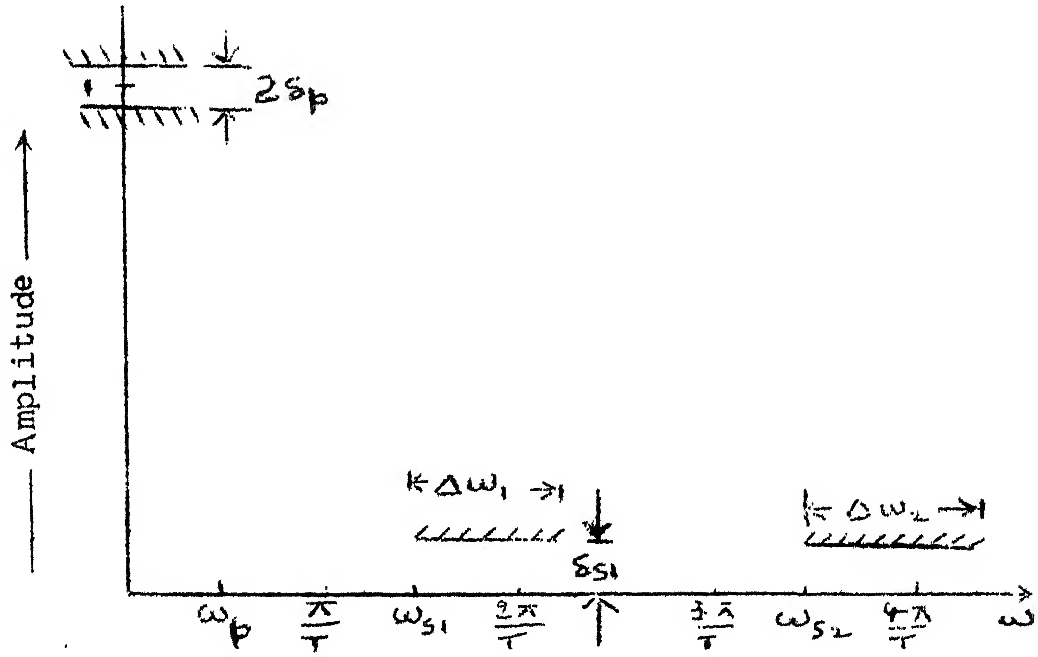


Fig. 4.3: Frequency response of LP filter

However in cases, where $\frac{\pi}{T}$ is significantly higher than the Nyquist frequency, the transition band between passband and stopband can be wider and it makes sense to define stopbands around each integer multiple of $\frac{2\pi}{T}$.

4.2.2 Order of the Filter:

The order of the lowpass filter mainly depends upon desired ripple specifications and cutoff frequencies. Given these specifications, the order of the filter can be calculated by the following relation [13,16]

$$N \approx \frac{D_\infty(\delta_p, \delta_s)}{\Delta F} - f(\delta_p, \delta_s)\Delta F + 1 \qquad (4.4)$$

where,

ΔF - Width of the transition band normalized to the sampling frequency.

$$D_\infty(\delta_p, \delta_s) = [5.309 \times 10^{-3}(\log_{10}\delta_p)^2 + 7.114\times10^{-2}(\log_{10}\delta_p) - 0.4761]\log_{10}\delta_s - [2.66\times10^{-3}(\log_{10}\delta_p) + 0.5941(\log_{10}\delta_p) + 0.4278] \quad (4.5)$$

$$f(\delta_p, \delta_s) = 0.51244\log_{10}(\delta_p/\delta_s) + 11.01217 \quad (4.6)$$

δ_p = tolerance in the magnitude response in the passband

δ_s = tolerance in the magnitude response in the stop band.

Generally, ΔF in (4.4) will be relatively small and the second two terms on the right side of (4.4) will be relatively insignificant compared to that of the first terms. Considering these approximations, equation (4.4) can be written as

$$N \cong \frac{D_\infty(\delta_p, \delta_s)}{\Delta F} \quad (4.7)$$

In Table 4.1, the function $D_\infty(\frac{\delta_p}{K}, \delta_s)$ is tabulated for some typical values of δ_p and δ_s and for K=1,2,..4.

Table 4.1

Tabulation of $D_\infty\left(\frac{\delta_p}{K}, \delta_s\right)$

δ_p	δ_s	K=1	K=2	K=3	K=4
0.100	0.0100	1.25	1.46	1.58	1.67
0.100	0.0050	1.41	1.63	1.75	1.84
0.100	0.0010	1.80	2.02	2.15	2.25
0.100	0.0005	1.95	2.19	2.32	2.42
0.100	0.0001	2.33	2.58	2.72	2.82
0.050	0.0100	1.46	1.67	1.79	1.88
0.050	0.0050	1.63	1.84	1.97	2.06
0.050	0.0010	2.02	2.25	2.38	2.47
0.050	0.0005	2.19	2.42	2.55	2.65
0.050	0.0001	2.58	2.82	2.96	3.06
0.010	0.0100	1.94	2.15	2.27	2.35
0.010	0.0050	2.12	2.33	2.45	2.54
0.010	0.0010	2.54	2.76	2.89	2.98
0.010	0.0005	2.72	2.94	3.07	3.16
0.010	0.0001	3.14	3.37	3.50	3.60
0.005	0.0100	2.15	2.35	2.47	2.55
0.005	0.0050	2.33	2.54	2.66	2.74
0.005	0.0010	2.76	2.98	3.10	3.19
0.005	0.0005	2.94	3.16	3.29	3.38
0.005	0.0001	3.37	3.60	3.73	3.83
0.001	0.0100	2.61	2.81	2.92	3.00
0.001	0.0050	2.81	3.01	3.12	3.20
0.001	0.0010	3.25	3.46	3.58	3.67
0.001	0.0005	3.45	3.66	3.78	3.87
0.001	0.0001	3.90	4.12	4.24	4.33

Referring to this table, the order of the filter is calculated directly for the tabulated values of δ_p and δ_s. Equation (4.4) has been realized in the subroutine for designing FIR filter in the simulation program. Given the cutoff frequencies for stopband and passband, δ_p and δ_s, the subroutine calculates the length of the filter and designs the filter for the calculated length.

The order of the filter, i.e., N should be always odd. Because if N is even then linear phase FIR filter will have a delay of atleast $\frac{1}{2}$ sample [17]. This half sample delay itself corresponds to interpolation between samples, thus such a filter can not preserve the samples of the original sequence. So N should be odd.

CHAPTER 5

WAVEFORM CODERS

Selection of a suitable waveform coding technique that encodes and decodes the sub-band signals keeping the bit-rate low, and maintaining the required quality of the output speech is important. The various waveforms coding techniques available are PCM, APCM, logarithmic companded PCM, DPCM, ADPCM, LDM, ADM etc.

Digital coding of Nyquist rated sampled signals is best accomplished using adaptive pulse code modulation (APCM) due to low sample-to-sample correlation of sub-band signals [1,3,4,5]. The adaptation logic and the various aspects of the selection of the coder, and the parameters of APCM coding for sub-bands have been discussed in Section 5.2.

One of the objectives of this thesis is to study whether the simplicity and adaptability of ADM coders can be exploited for encoding sub-band signals as an alternative to APCM coding. The adaptation logic and various aspects of ADM coding have been discussed in Section 5.3.

APCM coders and ADM coders fall under the category of adaptive quantizers and adaptive differential quantizers, respectively. The principles governing these quantizers have been discussed in the following section.

5.1 ADAPTIVE QUANTIZERS AND DIFFERENTIAL QUANTIZERS:

In quantizing speech signals, on the one hand, it is desired to keep the step size large enough to accomodate the maximum peak-to-peak range of the signal, on the other hand, it is desired to make the quantization step small so as to minimize the quantization noise,especially when the input signal amplitude is small. The amplitude of the speech signal can vary over a wide range depending on the speaker, the communication environment, and within a given utterance, from voiced to unvoiced segments. One approach to accomodate these amplitude fluctuations and maintaining a good SNR throughout is to use a non-uniform quantizer. An alternate approach is to adapt the properties of the quantizer to the level of the input signal. In the adaptation technique, a higher SNR can be obtained as compared to the non-uniform quantizer. Adaptive quantizers can be classified according to whether they are slowly or rapidly adapting, i.e. syllabic or instantaneous. The APCM and ADM coding, used in this work follow instantaneous adaptation as proposed by Jayant [1,19].

When adaptive quantization is used directly on samples of the input system, it is called adaptive pulse code modulation (APCM). The basic idea of adaptive quantization is

to let the step size (or in general the quantizer levels and ranges) vary so as to match the variance of the input signal. Adaptive quantization may be further classified as feed-forward and feedback adaptive quantizers as shown in Fig. 5.1. In feedforward adaptive quantizers, the amplitude

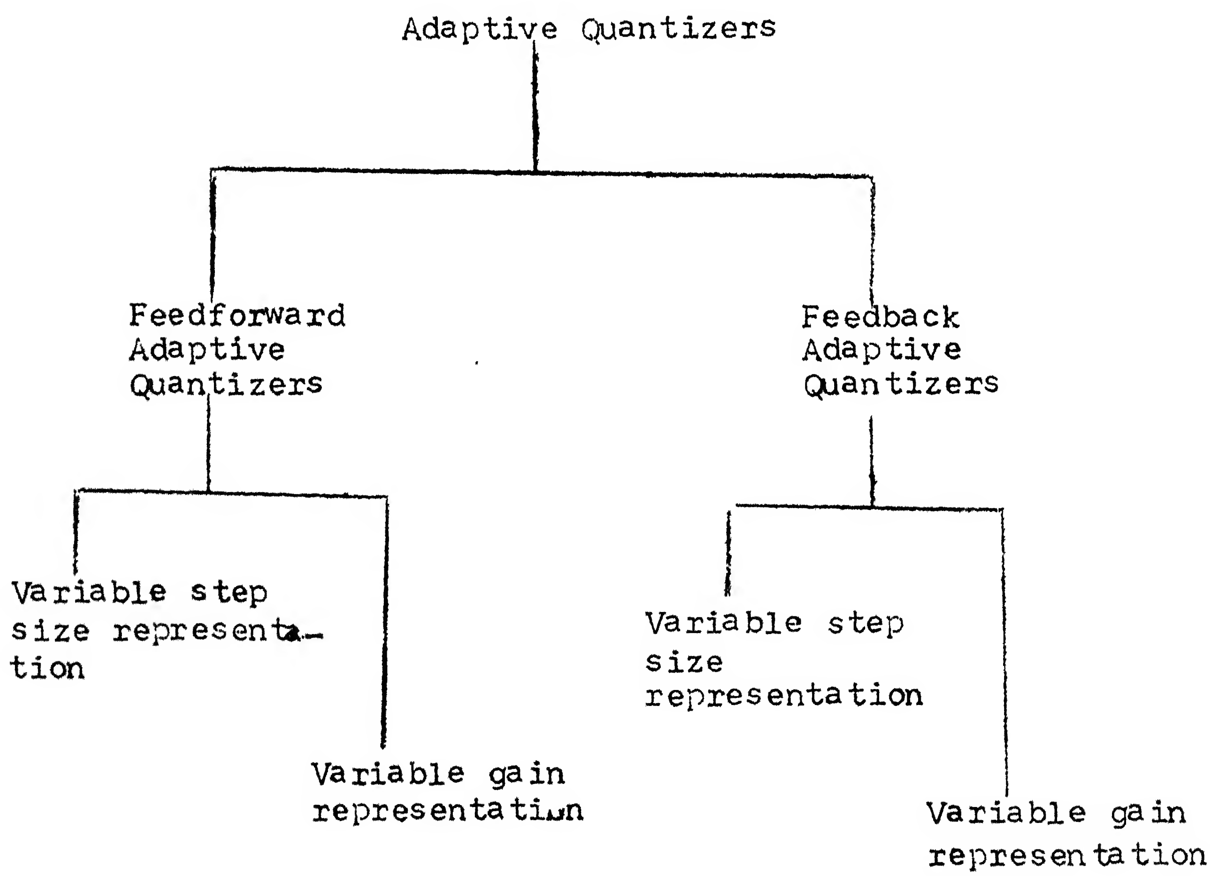


Fig. 5.1: Classification of Adaptive Quantizer

or variance of the input is estimated from the input itself. Whereas in feedback adaptive quantizers, the step size is

adapted on the basis of the output of the quantizer. The APCM coder used in this thesis follows feedback adaptation of the step size. This is illustrated in Figure 5.2.

Differential quantization scheme is motivated from the fact that there exists considerable correlation between adjacent speech samples. This correlation is significant even between the samples that are several sampling intervals apart. It suggests that the signal does not change rapidly from sample to sample so the difference between adjacent samples have a lower variance than the variance of the signal itself.

Figure 5.3 shows the general differential quantization scheme. The input to the differential quantizer is given by the equation

$$q(n) = x(n) - \hat{x}(n) \tag{5.1}$$

which is the difference between the unquantized input sample and its estimate. The estimated value is the output of a predictor system P. In differential quantizer, this difference signal is quantized rather than the input.

The quantized difference signal can be represented as

$$\hat{d}(n) = d(n) + e(n) \tag{5.2}$$

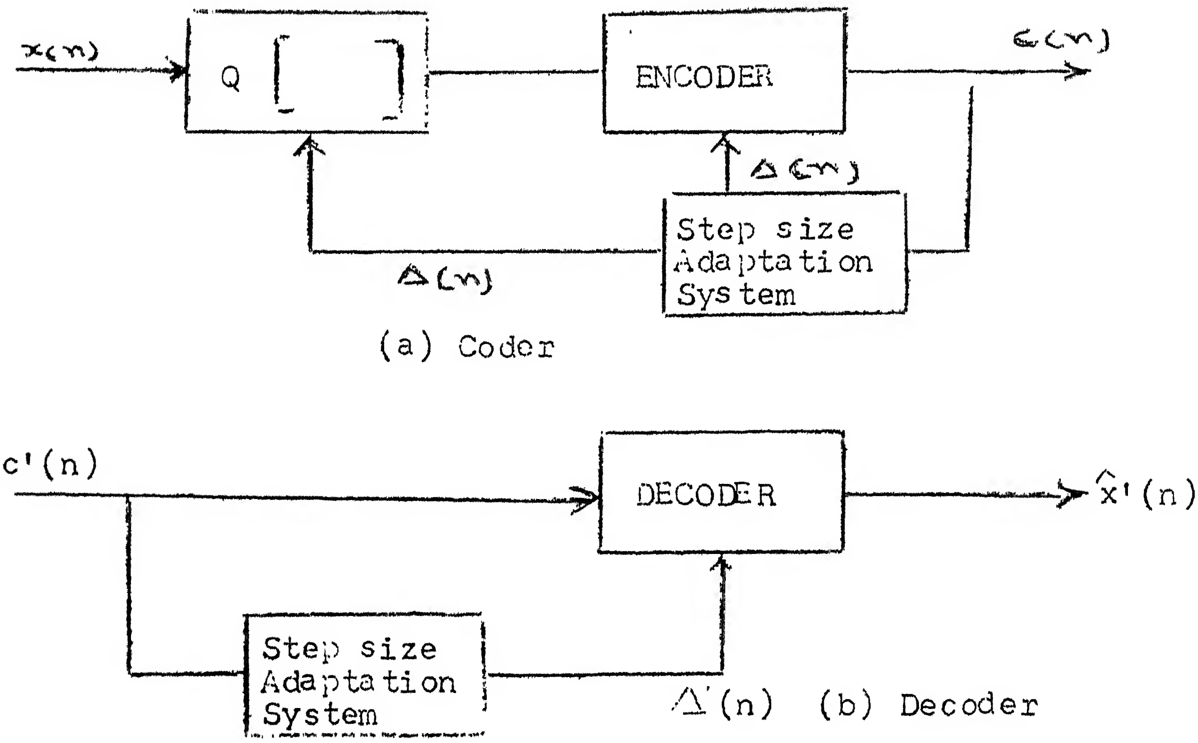


Figure 5.2: Feedback Adaptation of Step Size.

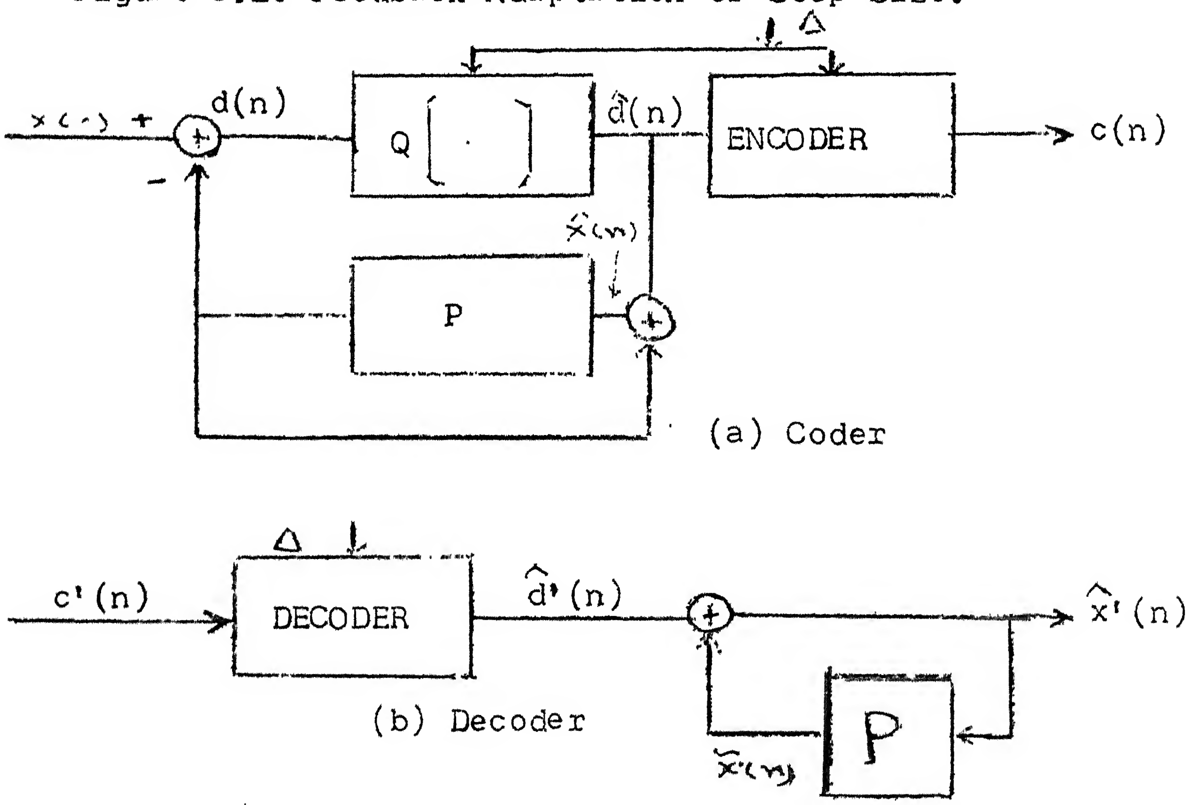


Figure 5.3: General Differential Quantization Scheme.

where e(n) is the quantization error. The quantized difference signal is added to the predicted value $\tilde{x}(n)$ to produce a quantized version of the input, i.e.

$$\hat{x}(n) = \tilde{x}(n) + \hat{d}(n) \tag{5.3}$$

Substituting equation (5.1) and equation (5.2) in equation (5.3) it is seen that

$$\hat{x}(n) = x(n) + e(n) \tag{5.4}$$

So it is quite evident that the quantized speech sample differs from the input only by the quantization error of the difference signal.

The simplest application of the concept of differential quantization is in delta modulation (DM). In DM system, the signal is over-sampled to increase adjacent sample correlation. At every sample, the sign of the difference between the input sample and the latest staircase approximation to it is transmitted and the next staircase approximation of input is increased or decreased by the step size depending upon the sign. The adaptive version of this technique is called adaptive delta modulation (ADM).

5.2 ADAPTIVE PULSE CODE MODULATION (APCM):

The choice of encoder parameter is determined in part by the static or long term spectral characteristic

of the speech waveform. Fig. 5.4 illustrates typical long-term speech spectra (averaged over a sentence) based on measurements made in reference [7,18]. Fig. 5.4 presents power spectral density of speech versus wraped frequency scale based on a constant (5 percent/division) contribution to the articulation index (AI) [7], in order to illustrate the relative perceptual importance of the various frequencies. Two possibilities for sub-band selection for low and high bit-rates have been illustrated. It is seen that across the entire speech spectrum there is characteristic drop in power density with increasing frequency. Across any one band, however, the drop in power density is relatively small. Since sub-band signals are low-pass translated and sampled at their Nyquist rate, they appear essentially as flat spectrum signals at the low sub-band sampling rate and have essentially no sample-to-sample correlation. Figure 5.5 shows examples of sub-band signals for band 1 to 4. Because of their low sample-to-sample correlation, encoding is best performed by adaptive PCM (APCM) [1,4].

5.2.1 Adaptation Logic and Coder Parameters for APCM Coder:

APCM uses adaptive quantization directly on input samples.

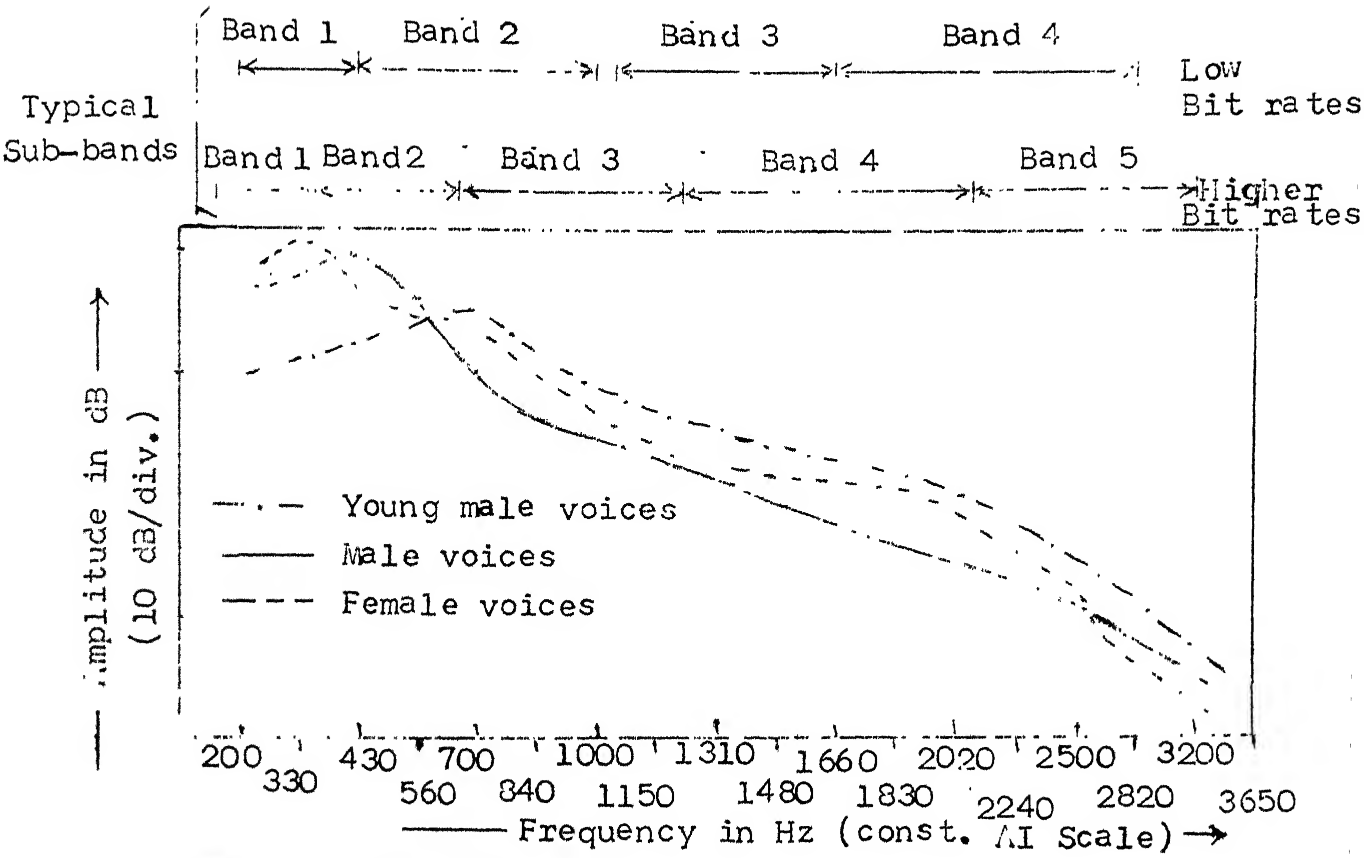


Figure 5.4: Long-term Spectrum of Speech.

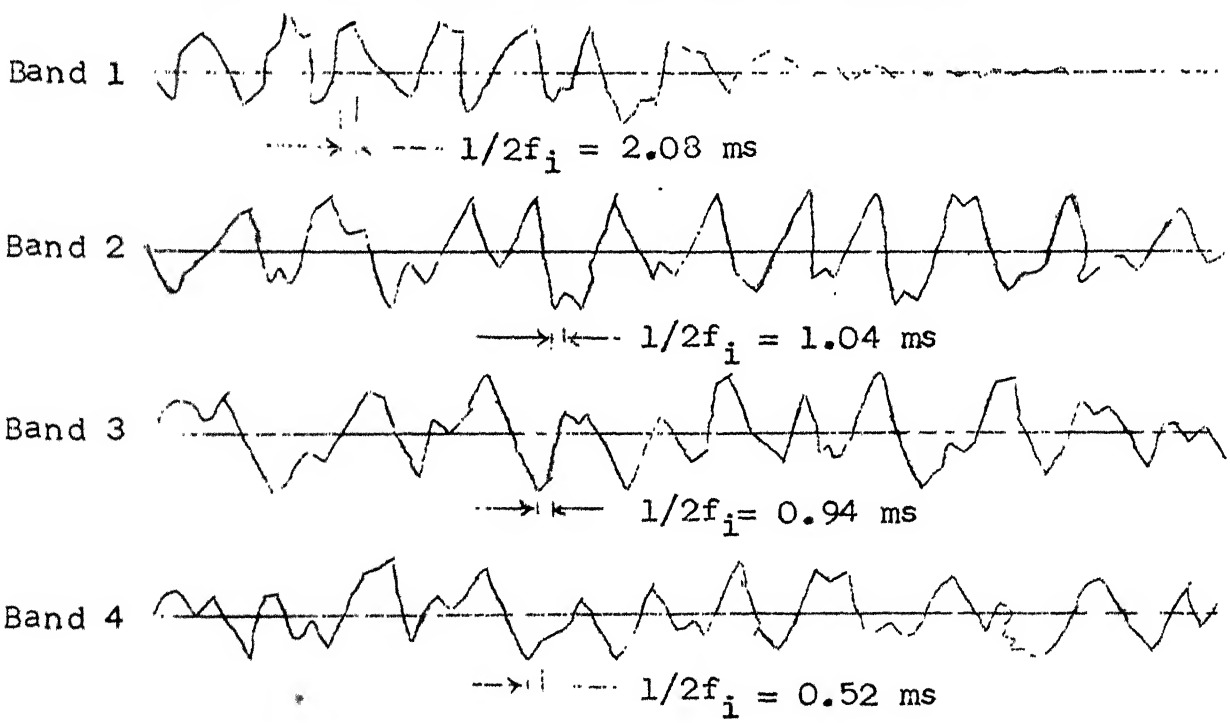


Figure 5.5: Typical Waveforms of Uncoded Sub-band Signals

The step size adaptation strategy used in simulation for the APCM coders is based on one-word step size memory approach [1]. The adaptation logic is shown in Fig. 5.6.

The coder input signal, denoted as X_r for rth sample, is quantized to one of the 2^B levels according to the quantizer characteristic as shown in Fig. 5.7, where B is the number of bits per sample. The step size adaptation circuit examines the quantizer output bits for the (r-1)th sample and computes the quantizer size, Δ_r for the rth sample according to the relation

$$\Delta_r = \Delta_{r-1} \cdot M(L_{r-1}) \qquad (5.5)$$

where,

$$\Delta_{MIN} \leq \Delta_r \leq \Delta_{MAX} \qquad (5.6)$$

and where Δ_{r-1} is step size used for (r-1)th sample. $M(L_{r-1})$ is a multiplication factor whose value depends on the quantizer magnitude level L_{r-1} at time (r-1). It can take on one of 2^{B-1} values, $M_1, M_2, \ldots, M_{2^{B-1}}$. If the lower magnitude quantizer levels are used at time (r-1), a value of $M(L_{r-1}) = M_i$ less than one is used to reduce the next step size. If upper magnitude levels are encountered, a value of M_i greater than one is chosen. In this way, the coder continuously adapts its step size in an attempt to track the short term variance of the input signal. The step

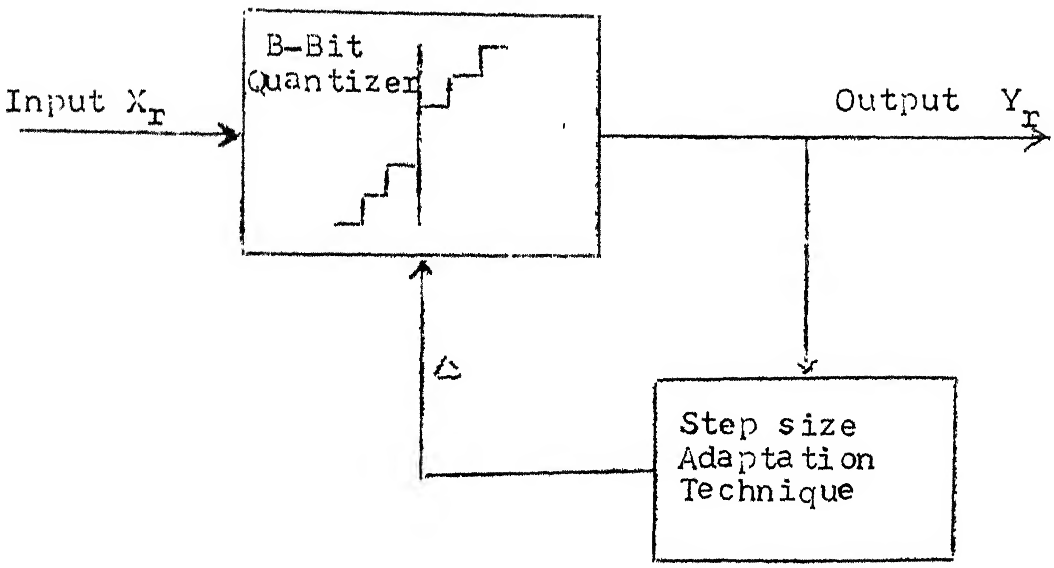


Figure 5.6: Step size Adaptation Algorithm.

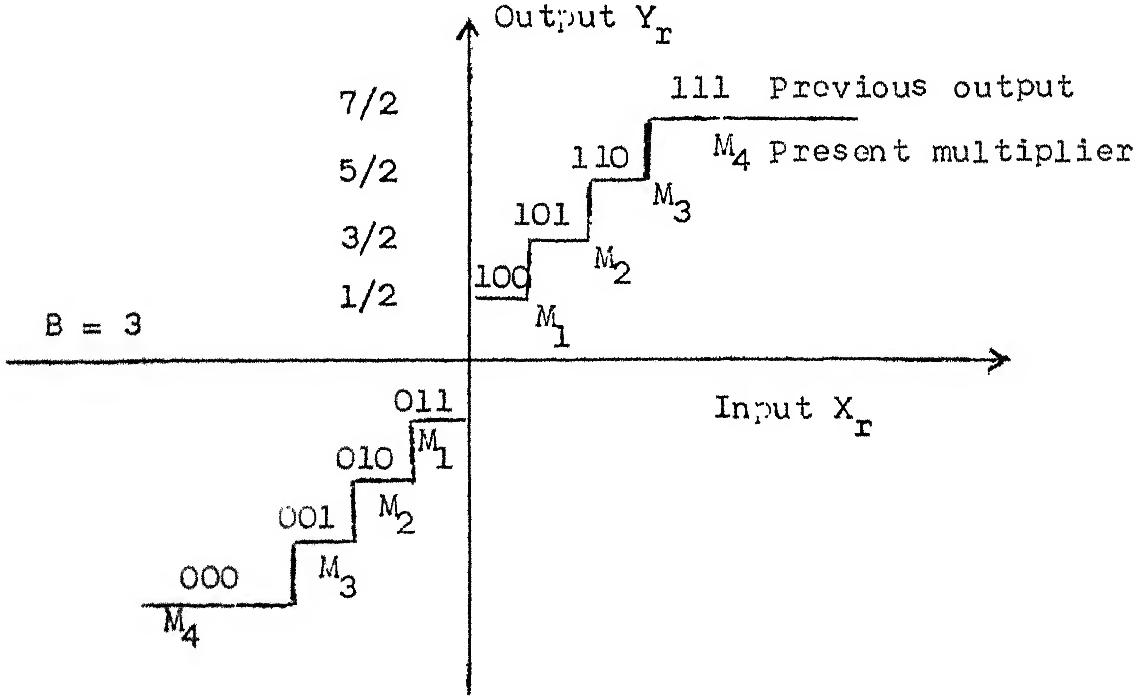


Figure 5.7: Quantizer Characteristic for APCM Coder.

size is always kept between some minimum and maximum value Δ_{MIN} and Δ_{MAX}, respectively. Typical values of M_i for 4,3,2, $1\frac{1}{2}$, $1\frac{1}{3}$, $1\frac{1}{4}$ bit APCM coders are given in Table 5.1 [1,6].

An interesting modification to the above algorithm has been proposed in reference [22]. This modification allows for encoding at an average bit rate of 1 +1/N bits/sample, where N is an integer. In this approach the sign of the signal X_r is encoded for each sample, r, and the magnitude of the signal is encoded with one bit every N samples. The step size adaptation is essentially that of equation (5.5) and the quantizer magnitude levels repeated for N-1 samples at the decoder. The sign bit transmits essentially the zero-crossing or phase information and the magnitude bit conveys the amplitude information in the waveform at a reduced rate.

With this modification, sampling rate of the order of 7.2 Kb/sec. can be achieved.

5.2.2 Dynamic Range of APCM Coder:

The dynamic range is the range of input variance, which the quantizer can handle. The quantities Δ_{MAX} and Δ_{MIN} in the above algorithm represent practical constraints in the adaptation logic. Their ratio determines the dynamic

TABLE 5.1

APCM CODER PARAMETERS

B =	4	3	2	$1\frac{1}{2}$	$1\frac{1}{3}$	$1\frac{1}{4}$
M_1	0.9	0.85	0.85	0.92	0.92	0.92
M_2	0.9	1.0	1.9	1.4	1.4	1.4
M_3	0.9	1.0				
M_4	0.9	1.5				
M_5	1.2					
M_6	1.6					
M_7	2.0					
M_8	2.4					

range that the coder can handle and their absolute values determine the centre of this dynamic range. For the input speech samples used in this work, a ratio of $\frac{\Delta_{MAX}}{\Delta_{MIN}} = 100$, was used, resulting in useful dynamic range of about 20 dB for the coder. The actual values of Δ_{MIN} and Δ_{MAX} must be different for each sub-band, to match properly the dynamic

range characteristics of the sub-band coder to that of the long term speech spectrum.

5.2.3 Allocation of Bits/Sample:

A useful measure for assisting in the parceling of bits among sub-bands is the signal-to-quantizing noise ratio (S/N) as a function of frequency [6]. Fig. 5.8 shows typical S/N values as a function of frequency that are found to give preferred signal quality at bit - rates of 16 and 9.6 Kb/sec., respectively. At 16 Kb/sec., it is found that good quality coding can be achieved with an allocation of 4 bits/sample ($\simeq$ 19 dB S/N) in lower sub-bands, 3 Bits/sample ($\simeq$ 12.75 dB) in middle sub-bands and 2 bits/sample ($\simeq$ 9 dB) in upper sub-bands. Similarly at the transmission rate of 9.6 Kb/sec allocation of 3 bits/sample in the lowest sub-band and 2 bits/sample in the remaining sub-bands is suggested.

5.3 ADAPTIVE DELTA MODULATION (ADM):

In order to exploit the simplicity of delta modulation at relatively low operating frequencies, adaptive delta modulator (ADM) has been proposed [19,20]. In ADM coding the variable step size increases during a step segment of input and decreases while quantizing a slowly varying segment of input waveform and thus step size follows

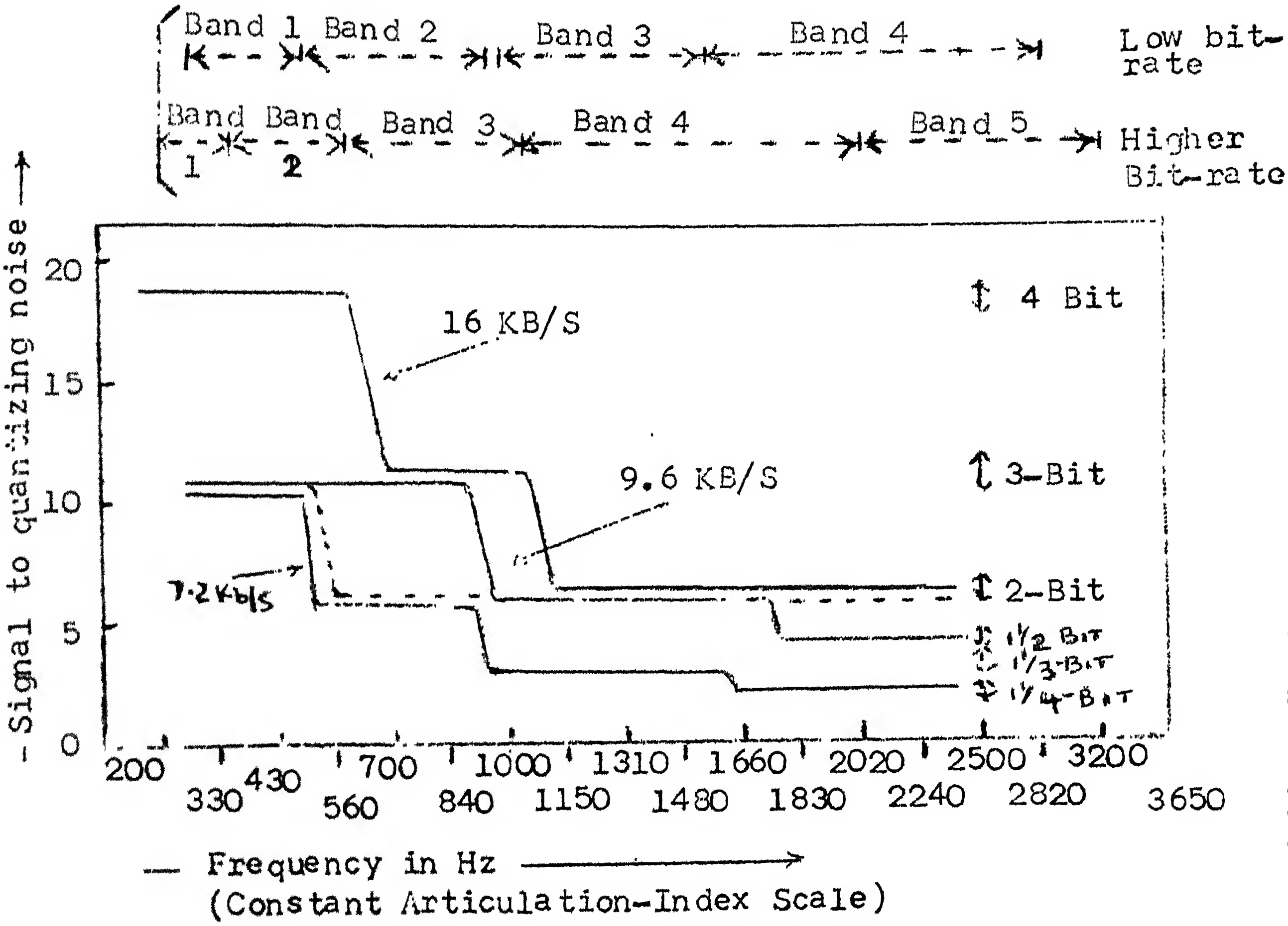


Figure 5.8: SNR vs. Frequency for Bit Allocation

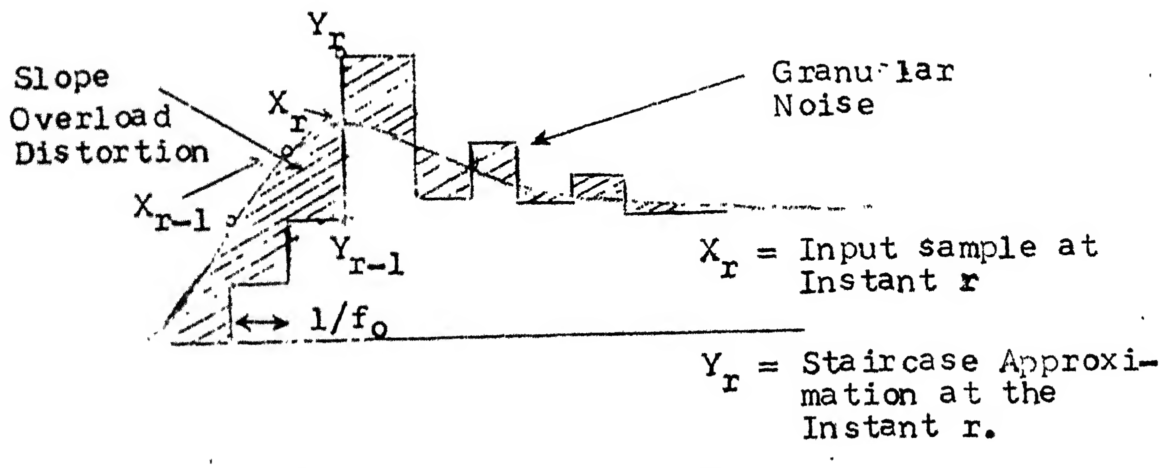


Figure 5.9: Adaptive Delta Modulation.

incoming bit C_r for a decision on the new step size m_r. Specifically, if the previous step size is denoted by m_{r-1}, the adaptation will be of the form

$$m_r = P.m_{r-1} \quad \text{if } C_r = C_{r-1} \;;$$

$$m_r = -Q.m_{r-1} \quad \text{if } C_r \neq C_{r-1} \qquad (5.7)$$

assuming that P and Q are time invariant.

Value of C_r is given by

$$S_{gn}\, m_r = C_r = S_{gn}\,(X_r - Y_{r-1}) \qquad (5.8)$$

where,

X_r is the amplitude of input signal and

Y_{r-1} is the amplitude of the latest staircase approximation to this at the sampling instant r.

5.3.2 Parameters of ADM Coders:

The crucial parameter of the adaptive delta modulator are time-invariant adaptation constants P and Q. The smallest and largest allowable step sizes are other important parameters.

In order to adapt to the signal during slope overload, it is necessary that

$$P > 1 \qquad (5.9)$$

and in order to converge to a constant input signal during a purely hunting situation ($m_r = Q\, m_{r-1}$), it is necessary that

$$Q < 1 \qquad (5.10)$$

Jayant [19] has shown that P and Q should satisfy the relation

$$PQ \leq 1 \qquad (5.11)$$

for stability, i.e., to maintain the step size at values appropriate for the level of the input signal. The values P = 1.5 and Q = 0.6666 have given good results in simulation for wide range of input speech samples [19].

Secondly, the step size limits should be chosen to provide the desired dynamic range for the input signal. The ratio of $\Delta_{MAX}/\Delta_{MIN}$ should be large enough to maintain a high SNR over a desired range of input signal levels. The minimum step size Δ_{MIN} should be as small as is practical so as to minimize the idle channel noise.

In the present work, the ratio of $\frac{\Delta_{MAX}}{\Delta_{MIN}}$ has been used equal to 100. The values of adaptation constants P and Q used are 1.5 and 0.6666 respectively.

CHAPTER 6

SIMULATION RESULTS AND CONCLUSION

In this thesis, simulation studies have been carried out on real speech samples for 9.6 and 16 Kb/s sub-band coders. Four band and five band sub-band coders have been designed for 9.6 and 16 Kb/s transmission rate respectively. Combinations of four, three and two bit quantization have been used in these designs.

The performance criterion used for simulation study, have been discussed in Section 6.1. Section 6.2 presents the results obtained from the simulation study. The details of the simulation program, input files and signal flowchart are presented in Appendix A. The conclusions have been drawn in Section 6.3.

6.1 PERFORMANCE CRITERION:

Two performance criterion have been used in simulation study. One is the conventional Signal-to-Noise ratio (SNR) and the other is segmental averaged SNR (SEGSNR).

6.1.1 Signal-to-Noise Ratio (SNR):

The long time averaged signal-to-noise ratio (SNR), in dB, is given by

$$SNR = 10 \log_{10} \left[\frac{\sum_{i=1}^{M} X_i^2}{\sum_{i=1}^{M} (\hat{X}_i - X_i)^2} \right] \tag{6.1}$$

where X_i, $i = 1,2,\ldots,M$ are the original signal samples and $\hat{X}_i$. $i = 1,2,\ldots,M$ are the corresponding reconstructed (processed) signal samples. M, the total number of samples processed, is 2048 in the present study.

In measuring the input and output signals to the sub-band coder, it is generally desirable to compensate for the effects of filtering in the coder [21]. This is done by an arrangement shown in Fig. 6.1.

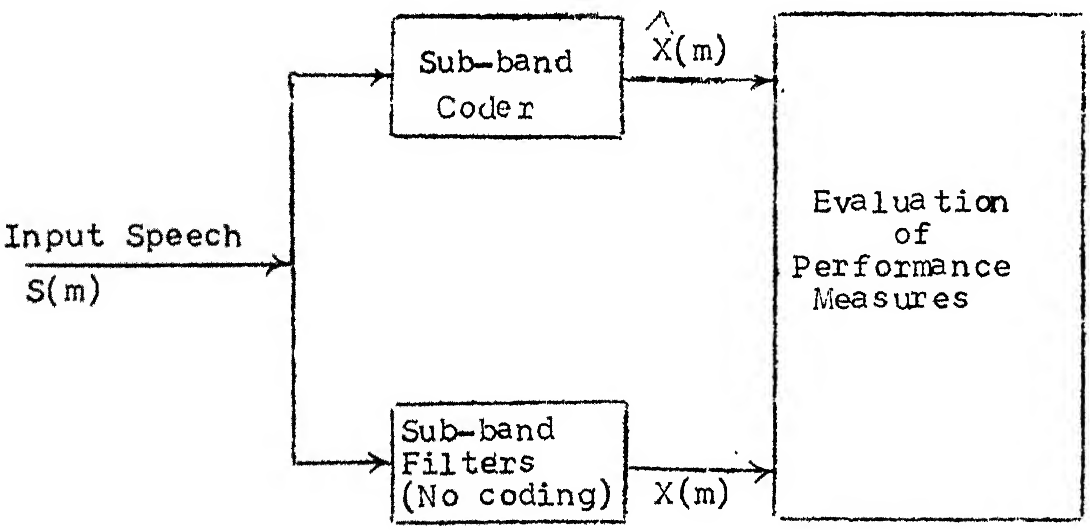


Fig. 6.1: Circuit for evaluating SNR of sub-band coder.

The input speech signal $S(m)$ is sub-band coded to form the output speech signal $\hat{X}(m)$. It is also filtered with the same filters used in the sub-band coder to generate a compensated reference signal $X(m)$, which is used as the input signal in equation (6.1). Thus, the SNR defined here is strictly a measure of coder distortions and is not affected by bandlimiting or group delay in the coder.

6.1.2 Segmental SNR (SNRSEG):

This is another performance measure being used now a days. It is believed to be more representative of the subjective quality of the output speech signal than the average SNR would indicate. Segmental averaged SNR, SNRSEG, is obtained by first calculating the SNR's (as defined in Section 6.1.1) for short segments and then averaging out these per-segment SNR's.

Let SNR(i) be the SNR of the ith segment in dB. Then,

$$\mathrm{SNRSEG} = \frac{1}{L} \sum_{i=1}^{L} \mathrm{SNR}(i) \qquad (6.2)$$

Where L is the total number of segments processed. The segment duration is typically taken as 20 ms [2]. A segment length of 128 samples has been used in the present simulation study. Then, $L = \frac{2048}{128} = 16$. To prevent any one segment

from dominating the average, the value of SNR(i) may be limited within a suitable range. The effect of threshold SNR(i) and the percentage of total segments on segmental SNR has also been studied.

6.2 SIMULATION EXPERIMENTS:

The simulation experiments were conducted on 2 sets of sub-bands for 9.6 Kb/s and 16 Kb/s transmission rates using APCM coders. These sets are given as follows:

SET No.	BAND EDGES IN Hz	
	From	To
1 (For 9.6 Kb/s transmission rate)	240	480
	480	960
	1067	1600
	1920	2880
2 (For 16 Kb/s transmission rate)	178	356
	296	593
	533	1067
	1067	2133
	2133	3200

Using these sets of sub-bands, overall signal-to-noise ratio (SNR) and segmental signal-to-noise ratio (SEGSNR) were measured. These results are discussed in Sub-section 6.2.1.

The dynamic range of the sub-band coder was also measured by varying signal input level for the whole system and individual APCM coders. These results are discussed in Sub-section 6.2.2.

ADM coders were also used in the simulation, for encoding sub-band signals, to study, whether the simplicity and adaptability of ADM coders can be exploited for encoding sub-band signals. These results are discussed in Sub-section 6.2.3.

6.2.1 Over-all and Segmental Signal-to-Noise Ratio:

The two sets of sub-bands, mentioned in Section 6.2 were used for simulation studies. The sub-band set was simulated,overall and segmental SNR's were measured. Signal-to-noise ratio of individual APCM coder of each sub-band was also measured. These results are tabulated in Table 6.1 and Table 6.2 for 9.6 Kb/s and 16 Kb/s transmission rates, respectively. The segmental SNR has been measured with 0 dB threshold of SNR(i). Fig. 6.2 shows the effect of threshold of SNR(i) on segmental SNR.

Table 6.1

9.6 Kb/s Four Band Sub-band Coder

Band	Decimate from 9.6 KHz	Band Edges in Hz From	Band Edges in Hz To	Sub-band sampling frequency (Hz)	Minimum step size Δ_{MIN}	Bit Allocation	Kb/s	Individual APCM Coder SNR (dB)
1	20	240	480	480	0.01125	3	1.44	13.33506
2	10	480	960	960	0.013	2	1.92	8.628801
3	9	1067	1600	1067	0.015	2	2.134	9.408046
4	5	1920	2880	1920	0.0155	2	3.84	8.444914
SYNC	-	-	-	-	-	-	0.266	
		Total Bit Rate Kb/s					9.6 Kb/s	
		OVERALL SNR (dB)					10.36130 dB	
		SEGMENTAL SNR (dB)*					8.589637 dB	

*Threshold = 0 dB

Table 6.2

16 Kb/s - 5-Band Sub-band Coder

Band	Decimate from 10.67 KHz	Sub-band Edge in Hz		Sub-band Sampling Rate (Hz)	Minimum Step size Δ_{MIN}	Bit Allocation	Kb/s	Individual APCM Coder SNR (dB)
		From	To					
1	30	178	356	356	0.00475	4	1.42	19.64087
2	18	296	593	593	0.0035	4	2.37	18.93022
3	10	533	1067	1067	0.007	3	3.20	12.16826
4	5	1067	2133	2133	0.013	2	4.27	9.220177
5	5	2133	3200	2133	0.013	2	4.27	9.732607
SYNC	-	-	-	-	-	-	0.47	-

TOTAL BIT RATE (Kb/s) = 16 Kb/s

OVERALL SNR = 13.917365 dB

SEGMENTAL SNR* = 16.79234 dB

*Threshold = 0 dB

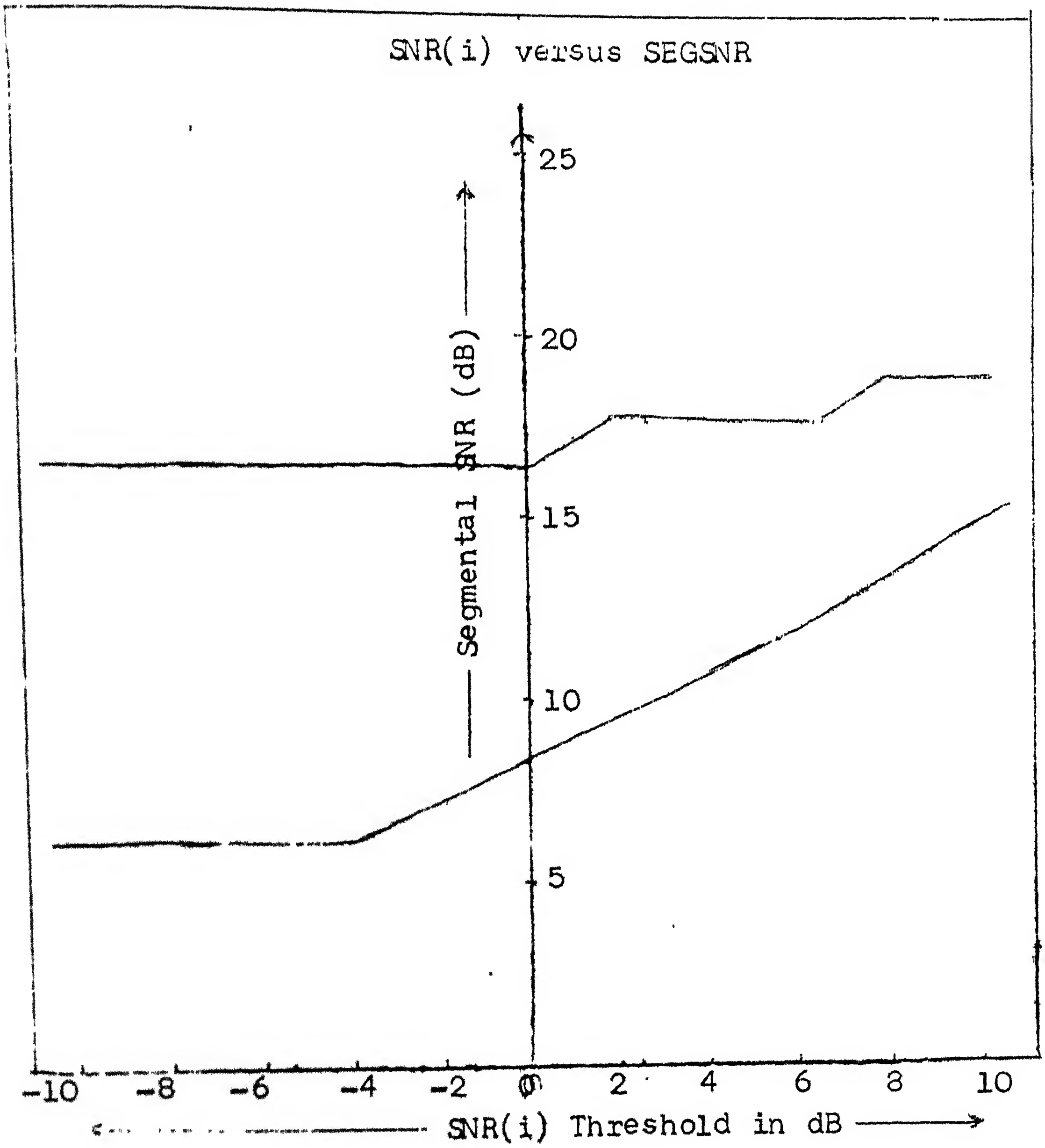


Figure 6.2: Effect of Threshold SNR(i) on SEGSNR.

Table 6.3

Variation of Segmental SNR with SNR(i) Threshold for 9.6 Kb/s Four-Band Sub-band Coder.

SNR(i) (dB)	Percentage of segments* used	SEGSNR (dB)
-10	93.75	5.801747
- 3	87.5	6.435395
- 2	81.25	7.113296
- 1	75	7.846380
0	68.75	8.589637
3	62.5	11.28653
6	56.25	12.18344
7	50	14.57595
10	43.75	15.87721

*Total number of segments = 16

Table 6.4

Variation of Segmental SNR with SNR(i) Threshold for 16 Kb/s Five-band Sub-band Coder.

SNR(i) (dB)	Percentage of segments* used	SEGSNR (dB)
-10	93.75	16.79234
- 6	93.75	16.79234
- 2	93.75	16.79234
0	93.75	16.79234
+ 2	87.5	17.90
+ 4	87.5	17.90
+ 6	87.5	17.90
+ 8	81.25	19.6467930
+10	81.25	19.6467930

*Total number of segments = 16

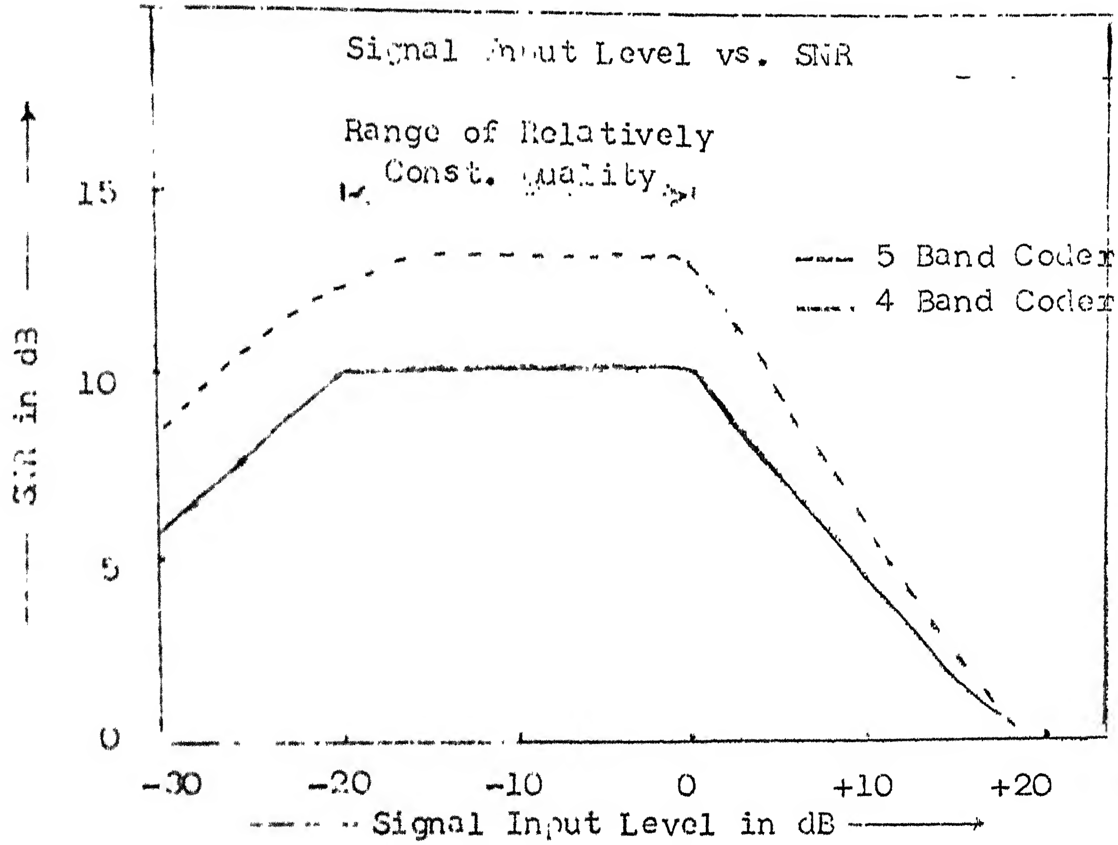


Figure 6.3: Dynamic Range of Sub-band Coder.

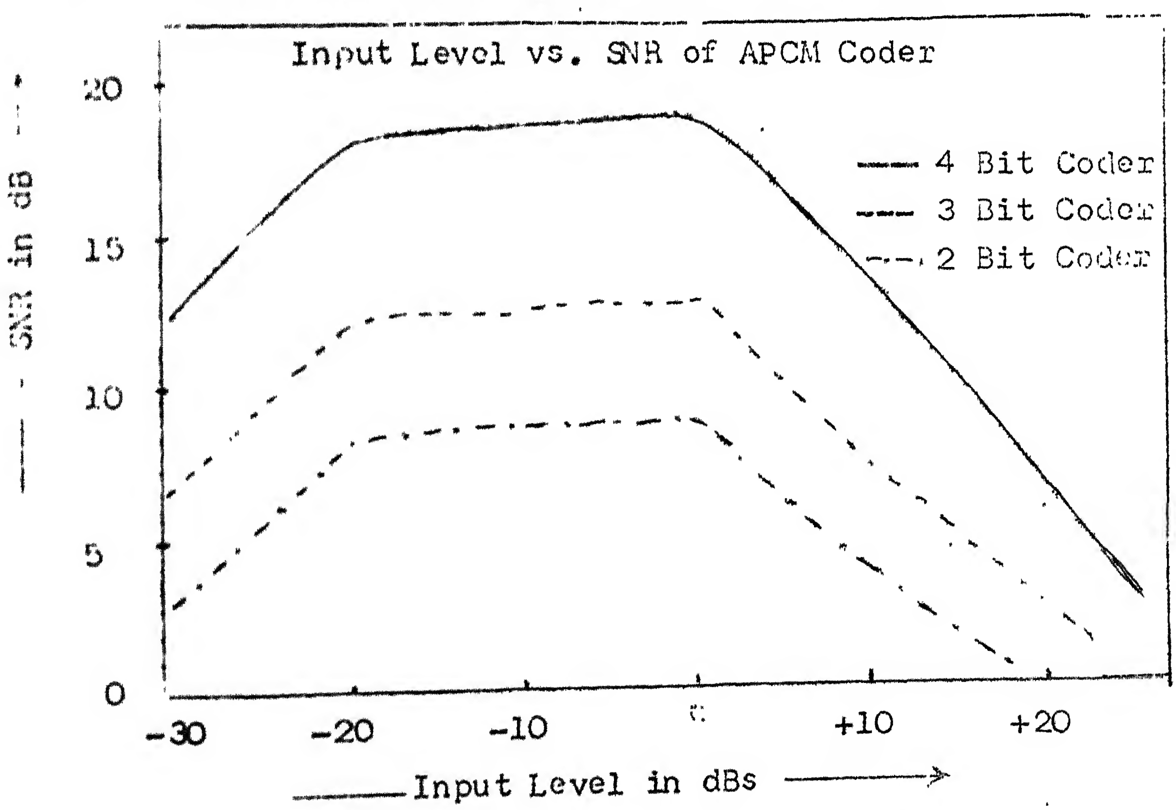


Figure 6.4: Dynamic Range of Individual APCM Coder.

Table 6.3 and Table 6.4 show the variation of segmental SNR with the variation in SNR(i) threshold and the corresponding percentage of segments used.

6.2.2 Dynamic Range Measurements:

The dynamic range of the sub-band coder and individual APCM coders was assessed by varying input levels from 0 dB to -30 dB and 0 dB to + 20 dB. The results are shown in Fig. 6.3 for sub-band coder and in Fig. 6.4 for individual 4-bit, 3 bit and 2 bit APCM coders.

6.2.3 Performance of ADM Coder:

As an alternative to APCM coder, ADM coder was used for encoding sub-band signals. The sub-band Nyquist rate ($f_s = 2f_i$) was increased by 5,10,15 and 20 times, the average SNR was measured in each case. The results obtained are listed in Table 6.5.

Table 6.5

Performance of ADM Coder

S.No.	Number of times Sub-band Nyquist sampling rate increased	SNR (dB)
1	20	9.6398220
2	15	6.7524877
3	10	4.019430
4	5	0.7624895

6.3 CONCLUSIONS:

Sub-band coding offers a new alternative in speech encoding which fills the gap between waveform coding techniques and vocoding techniques. More generally, it is related to a class of coders known as transform coders [3]. But it is a simpler alternative to the complex transform coders. It offers a quality that is significantly better than conventional waveforms coders at low bit-rates.

The design of sub-band coder involves the consideration of a large number of parameters and trade-offs. For many of these parameters, no analytical means exist for choosing them in an optimal way. In this thesis it is attempted to provide some guidelines and insight for selecting parameters of sub-band coders. These guidelines are based upon the computer simulation work carried out.

The following conclusions can be drawn from the simulation study undertaken.

a) Linear phase FIR filter is a better choice for bandpass filters, used for partitioning of the speech band and for avoiding aliasing in sampling rate conversion.

b) Filter length of the order of 175-200 taps is sufficient for partitioning of each sub-band of the speech band.

c) Individual bandpass filters must be used for each sub-band, in place of a filter bank, for easier implementation.

d) A transition width of the order of 50 Hz in bandpass filter is sufficient.

e) A passband ripple of 0.173 dB and stopband attenuation of 46 dB are good compromises for filter tolerance specification.

f) The Nyquist rated sub-band signals have very poor sample-to-sample correlation, hence their encoding is best accomplished by APCM coder. For the same reason, differential coders do not perform well in a sub-band coder system.

g) Signal-to-noise ratio increases with number of bits allocated per sample.

h) When maximum-to-minimum step-size ratio of APCM coder is 100, and the minimum step size is chosen to give maximum SNR, the quality of the coder remains relatively constant over a range of input levels of about 20 dB, which is about 10 dB less than the range reported by Crochiere [6,21].

i) The overall SNR for 9.6 and 16 Kb/s coders have been found to be 10.36130 and 13.917365 dBs, which are quite close to the SNR's reported by Crochiere [6,21] for the same transmission rates.

j) The average values of SNR for 4,3 and 2 bit quantizers have been found to be 19, 12.75 and 9 dBs, which are quite close to the SNR's reported by Crochiere [6].

k) Adaptive delta modulation requires a sampling rate on the order of 10 times or more of the Nyquist rate for good performance. So it seems that it is not suitable for encoding sub-band signals, keeping the bit-rate low [12].

l) The simulation study has been carried out with only one set of speech samples consisting of 2048 samples. For a better critical assessment of the performance of the sub-band coder, this study should have been carried out with different sets of speech samples consisting of a larger number of samples. The average results obtained in such a case would have been a better indicator of the performance of the system.

6.4 APPLICATION:

There is an increasing need of efficient waveform coders in the 7.2 to 16 Kb/s range to benefit from

the possibility of digital speech transmission over voice band provided by switched telephone lines and mobile VHF radios. Sub-band encoder promises to fulfil this need.

Other potential applications of sub-band coders are in the field of narrow band communication, voice storage application, voice coordination on digital data lines and for secure voice communications by digital encryption and transmission over conventional data lines.

6.5 SUGGESTION FOR FUTURE WORK:

The following investigations may be carried out as follow-up of the present simulation study.

a) Subjective assessment of coder quality is important. As such extensive subjective tests could be carried out to gather data to supplement the SNR figures documented in this thesis. Different sets of speech samples at various sampling rates could be used for this purpose.

b) The transmission rate of sub-band coder can be brought down to 7.2 Kb/sec. by implementing modification to APCM logic as suggested in Chapter 5 [22].

c) The transmission bit rate of the sub-band coder may be further reduced to 4.8 Kb/s by implementing variable-band coding scheme for speech encoding suggested in reference [23].

d) Hardware implementation using a general purpose signal processing chip like INTEL 2920 can be investigated.

REFERENCES

[1] N.S. Jayant, 'Digital coding of speech waveforms: PCM, DPCM and DM quantizers,' Proc. IEEE, vol. 62, pp. 611-632, May 1974.

[2] J.L. Flanagan, et.al., 'Speech coding,' IEEE Trans. Commun., vol. COM-27, pp. 710-737, Apr. 1979.

[3] P. Noll, 'Transform coding of speech,' ICC 77, vol. 1, pp. 306-309, June 1977.

[4] R.E. Crochiere and J.L. Flanagan, 'Sub-band encoding of speech,' ICC 77, vol. 1, pp. 293-296, June 1977.

[5] R.E. Crochiere, S.A. Webber and J.L. Flanagan, 'Digital coding of speech in sub-bands,' Bell Syst. Tech. Jr., vol. 55, pp. 1069-1085, Oct. 1976.

[6] R.E. Crochiere, 'On the design of the sub-band coders for low bit rate speech communication,' Bell Syst. Tech. J., vol. 56, pp. 747-770, June 1977.

[7] L.L. Beranek, 'The design of communication systems,' Proc. IRE 35, pp. 880-890, Sept. 1947.

[8] K.D. Kryster, 'Method for calculation and use of articulation index,' J. Acoust. Soc. America, 34, pp. 1689-1998, 1962.

[9] H. Flatcher, J.C. Steinberg, 'Articulation testing methods,' Bell Syst. Tech. J., vol. 8, pp. 806-854, Oct.1929.

[10] J.H. McClellan, T.W. Parks and L.R. Rabiner, 'A computer program for designing optimum FIR linear phase digital filter,' IEEE Trans. Audio Electroacoust., vol. AU-21, pp. 506-525, Dec. 1973.

[11] Abraham Peled, Bede Lie, 'Digital signal processing, theory, design and implementation,' John Wiley and Sons, Inc., New York 1976.

[12] L.R. Rabiner and R.W. Schafer, 'Digital processing of speech signals,' Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1978.

[13] R.E. Crochiere, L.R. Rabiner, 'Optimum FIR digital filter implementations for decimation, interpolation and narrow band filtering,' IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-23, No. 5, pp. 444-456, Oct. 1975.

[14] R.W. Schafer, L.R. Rabiner, 'A digital signal processing approach to interpolation,' Proc. IEEE, vol. 61, pp. 692-702, June 1973.

[15] R.E. Crochier, 'A general program to perform sampling rate conversion of data by rational ratios,' Programs for digital signal processing, IEEE Press, The Institute of Electrical and Electronics Engineers, Inc., New York, pp. 8.2-1 to 8.2-7, 1979.

[16] O. Herrmann, L.R. Rabiner and D.S.K. Chan, 'Practical design rules for optimum FIR low pass filter,' Bell Syst. Tech. J., vol. 52, pp. 789-799, Aug. 1973.

[17] L.R. Rabiner and R.W. Schafer, 'Recurrsive and non-recurrsive realization of digital filters designed by frequency sampling techniques,' IEEE Trans. Audio, Electroacoust., vol. 19, pp. 200-207, Sept. 1971.

[18] H.K. Dunn and S.D. White, 'Statistical Measurements on conversational speech,' J. Acoust. Soc. Amer., (11), pp. 278-288, 1940.

[19] N.S. Jayant, 'Adaptive Delta Modulation with one-bit memory,' Bell Syst. Tech. J.,vol. 49, pp. 321-342, March 1970.

[20] J.E. Abate, 'Linear and adaptive delta modulation,' Proc. IEEE, vol. 55, No. 3, pp. 298-307, March 1967.

[21] R.E. Crochiere, 'An analysis of 16 Kb/s sub-band coder performance: dynamic range, tandem connections, and channel errors,' Bell Syst. Tech. J., vol. 57, Oct. 1978.

[22] D.J. Goodman and R.M. Wilkinson, 'A robust adaptive quantizer,' IEEE Trans. on Communication, vol. COM-23, pp. 1362-1365, November 1975.

[23] R.E. Crochiere and M.R. Sambur, 'A variable-band coding scheme for speech encoding at 4.8 Kb/s', Bell Syst. Tech. J., vol. 56, No. 5, pp. 771-779, June 1977.

APPENDIX A

SUB-BAND CODER SIMULATION

Non-real time simulation of the sub-band coder has been carried out on DEC SYSTEM 1090 at IIT Kanpur.

The simulation program has been written in FORTRAN-IV language. Some features of FORTRAN 10 language have also been used. FORTRAN language is preferred because it is supported by most computer system and it supports complex arithmatic.

The program consists of the main program and several subroutines and functions. The main program calls the subroutine in proper order. The program is designed to be interactive. The flowchart of the simulation program is presented in Fig. A.1. A full listing of the program is given in Appendix C.

The various input files have been discussed in Section A.1.

A.1 INPUT FILES:

The various input data required by the program in various subroutines is provided by input files. There are three different input files; for designing bandpass filters, for sampling rate conversion on the coder side and for

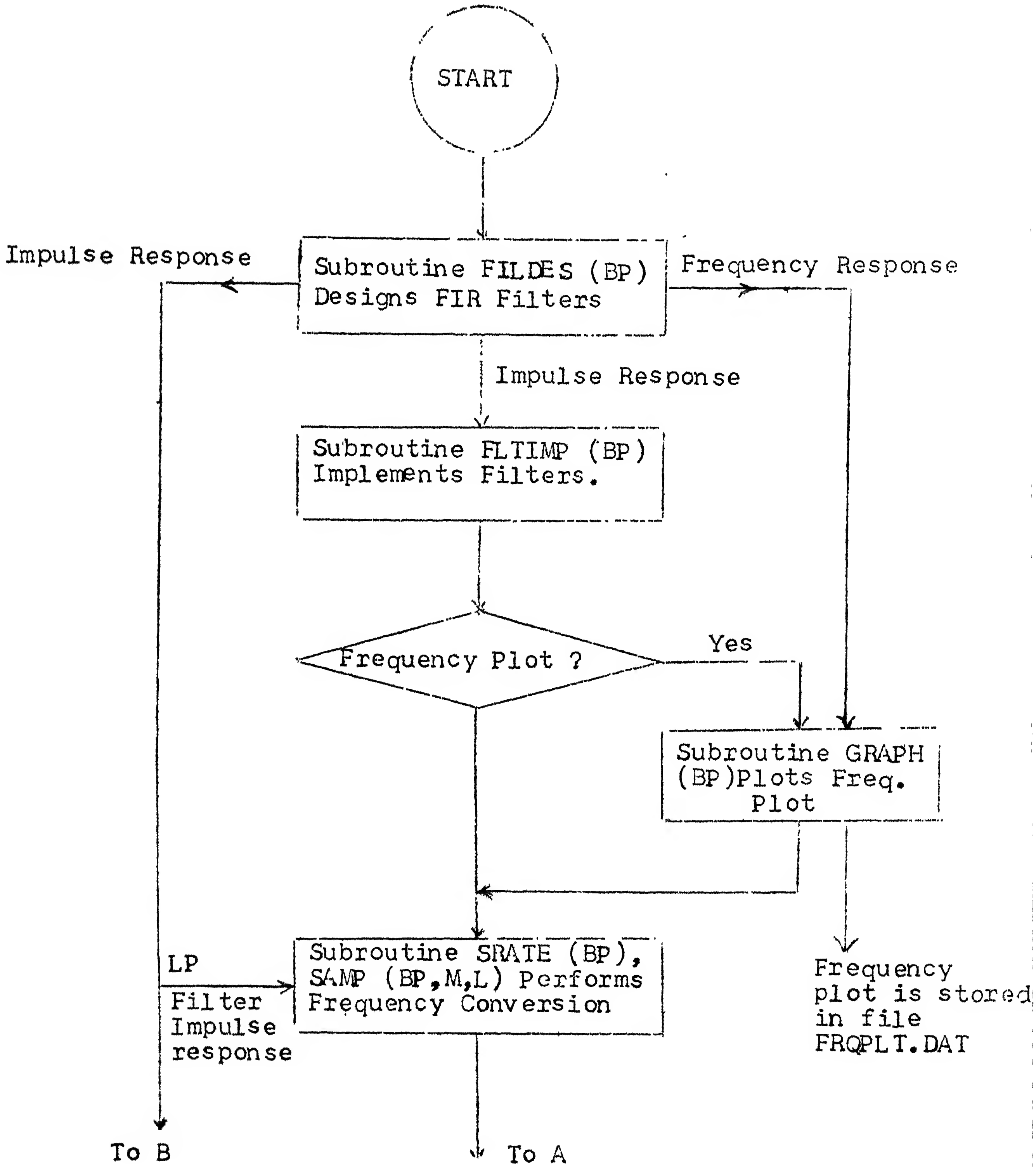

START
Impulse Response
Subroutine FILDES (BP)
Designs FIR Filters
Frequency Response
Impulse Response
Subroutine FLTIMP (BP)
Implements Filters.
Frequency Plot ?
Yes
Subroutine GRAPH
(BP)Plots Freq.
Plot
LP
Subroutine SRATE (BP),
SAMP (BP,M,L) Performs
Frequency Conversion
Filter
Impulse
response
Frequency
plot is stored
in file
FRQPLT.DAT
To B
To A

Fig. A.1 contd.

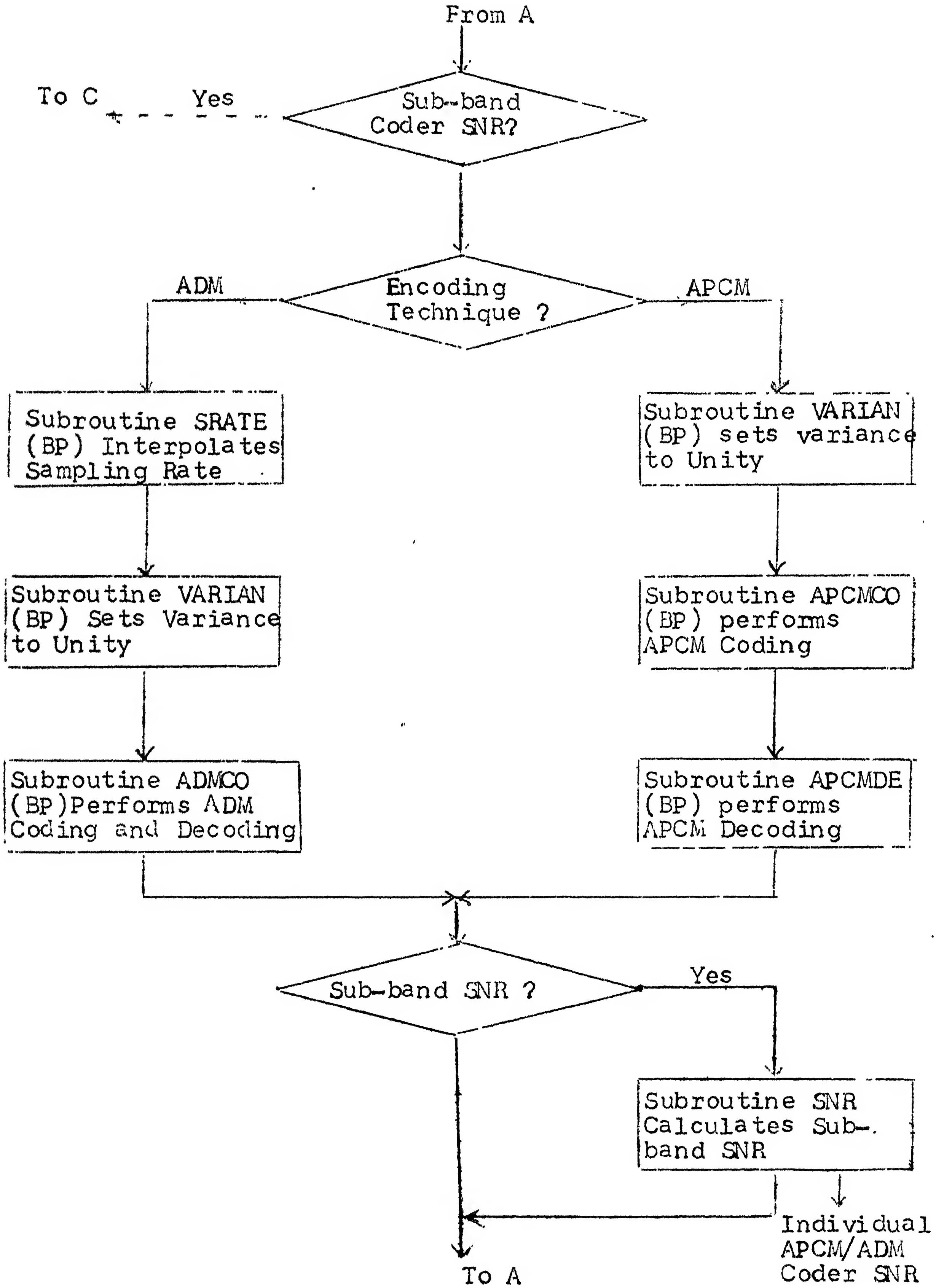

From A
Sub-band Coder SNR?
Yes
To C
Encoding Technique ?
ADM
APCM
Subroutine SRATE (BP) Interpolates Sampling Rate
Subroutine VARIAN (BP) Sets Variance to Unity
Subroutine ADMCO (BP)Performs ADM Coding and Decoding
Subroutine VARIAN (BP) sets variance to Unity
Subroutine APCMCO (BP) performs APCM Coding
Subroutine APCMDE (BP) performs APCM Decoding
Sub-band SNR ?
Yes
Subroutine SNR Calculates Sub-band SNR
Individual APCM/ADM Coder SNR
To A

Fig. A.1 contd..

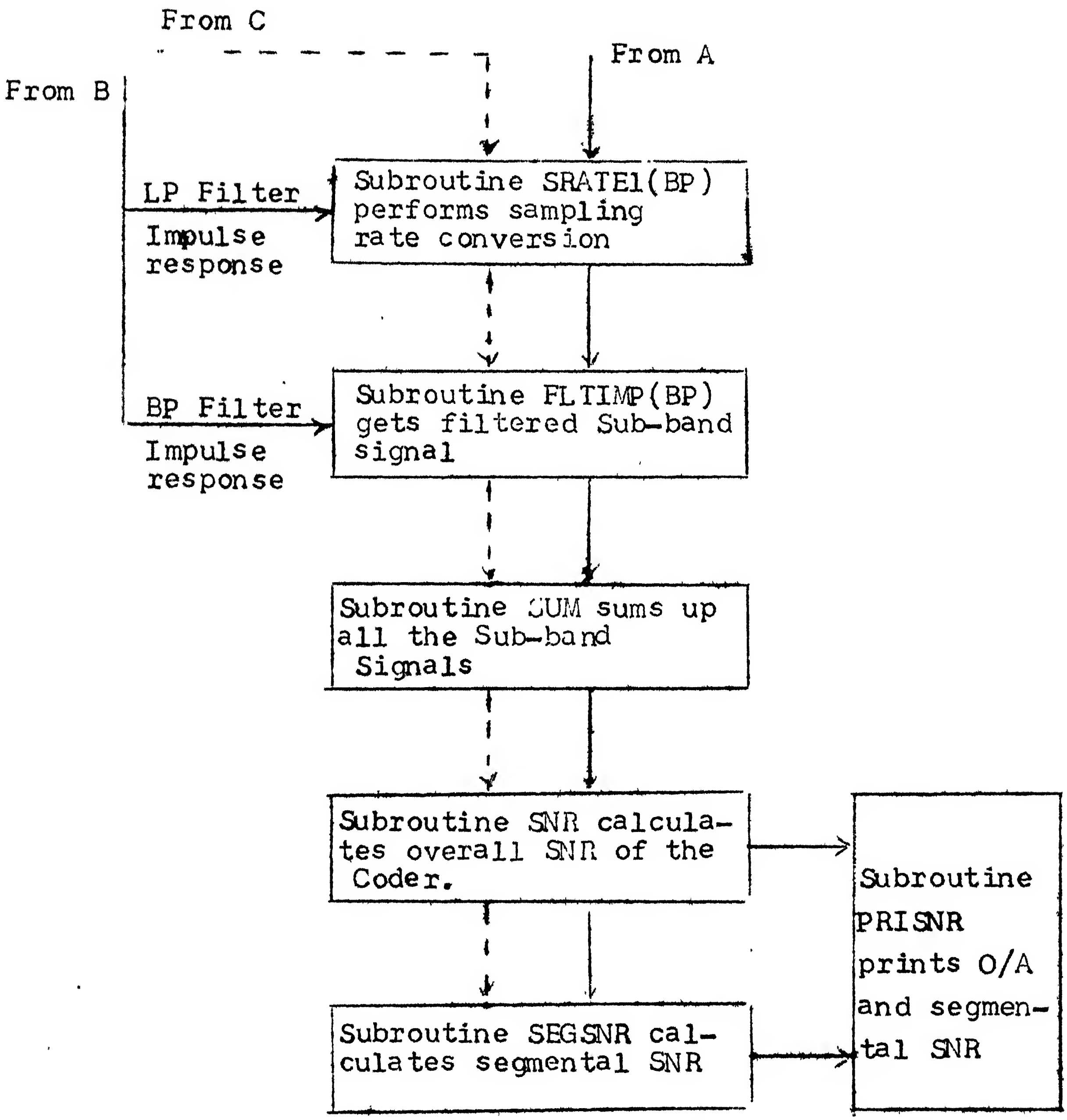


Figure A.1: Flowchart of Sub-band Coder Simulation Program.

sampling rate conversion on the receiver side. These files have to be amended after considering various issues of design discussed in this thesis, before processing sub-bands. These files are shown in Appendix A.1. The details of various data supplied by these files are discussed as follows.

Input File for Designing Bandpass Filter:

As an example first line specifies that a 175 tap FIR filter is to be designed with 3 bands and 0 grid density. The second line specifies that the desired amplitude response is 1 in pass-band and 0 in stop band. The third line specifies that the errors in passband and stopband should be weighed by 1 and 10, respectively. Line number 5 to 8 specify band edges for the required passbands and stopbands. These input frequencies are normalized to 0.5. In the first line the length of the filter is indicated as 0, if the lowpass filter is to be designed. In this case, the subroutine calculates the length of the filter by equation (4.4) of Chapter 4 and designs the filter accordingly.

Input File for Sampling Rate Conversion on the Coder Side:

This file can be divided into segments of 7 line each. Each segment provides input data for one stage decimation/interpolation. If the sampling rate conversion is

to be done by a factor of more than 15, then two stage interpolation/decimation is used.

In each segment, the first line indicates sub-band number, the second line indicates values of M and L. M is the integer by which sampling rate has to be decimated and L is the integer by which the sampling rate has to be increased. Line number 3 to 5 provide the lowpass filter specifications as explained earlier. The last line of the segment gives the ripple in passband and attenuation in stopband in dBs.

Input File for Sampling Rate Conversion on the Receiver Side:

This file is divided into segment of 2 lines each. The first line indicates the sub-band number and the second line indicates the values of M and L.

APPENDIX A.1

INPUT FILES FOR SIMULATION PROGRAM

Input Files for Designing Bandpass Filters

Line No.	Input Data
1	175 3 0
2	0 1 0
3	10 1 10
4	
5	0.0 0.025 0.0302083 0.04479 0.05 0.5
6	0.0 0.05 0.00552083 0.0947916 0.1 0.5
7	0.0 0.1111438 0.1166667 0.16144583 0.16666 0.5
8	0.0 0.2 0.2052083 0.2947917 0.3003125 0.5

Input File for Sampling Rate Conversion on the Coder side.

Segment No.	Line No.	Input Data
1	1	1
	2	5 1
	3	0 2 0
	4	1 0
	5	1 10
	6	0.0 0.0927083 0.1 0.5
	7	0.173 46

Segment No.	Line No.	Input Data
2	1	1
	2	4 1
	3	0 2 0
	4	1 0
	5	1 10
	6	0.0 0.0177083 0.025 0.5
	7	0.173 46
3	1	2
	2	10 1
	3	0 2 0
	4	1 0
	5	1 10
	6	0.0 0.0427083 0.05 0.5
	7	0.173 46
4	1	3
	2	9 1
	3	0 2 0
	4	1 0
	5	1 10
	6	0.0 0.0482292 0.0555208 0.5
	7	0.173 46

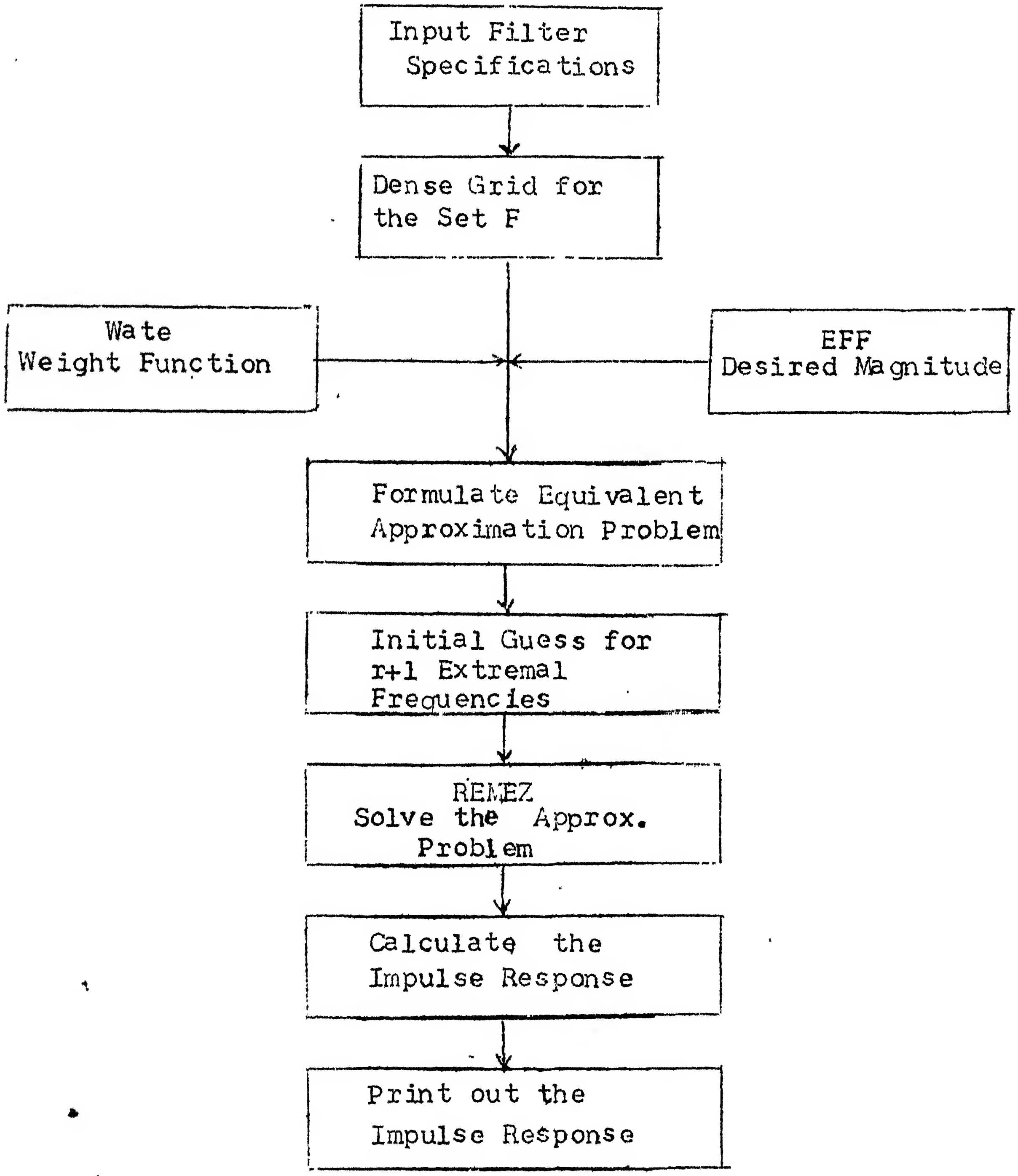
Segment No.	Line No.	Input Data
5	1	4
	2	5 1
	3	0 2 0
	4	1 0
	5	1 10
	6	0.0 0.092708 0.1 0.5
	7	0.173 46

Input File for Sampling Rate Conversion on the Receiver side:

Segment No.	Line No.	Input Data
1	1	1
	2	1 4
2	1	1
	2	1 5
3	1	2
	2	1 10
4	1	1
	2	1 9
5	1	4
	2	1 5

APPENDIX B

FILTER DESIGN FLOWCHART



APPENDIX C

PROGRAM LISTING

The following pages contain a listing of the Sub-band coder simulation program in FORTRAN.

**

```
C-----------------------------------------------------------------------
C
C                    MAIN PROGRAM FOR SUB BAND CODER.
C        THIS IS THE MAIN PROGRAM FOR SIMULATING SUB-BAND CODER.
C        THE VARIOUS SUB-ROUTINES ALONG WITH THEIR INPUT/OUTPUT
C        PARAMETERS ARE EXPLAINED AS THEY OCCUR IN THE PROGRAM.
C
C-----------------------------------------------------------------------

        COMMON/A4/NFILT,HP
        COMMON/A5/HP1,KK
        COMMON/A8/NN,AVAL
        COMMON/A9/SNRDB
        DIMENSION HP(5,12,404),HP1(5,2050),HP3(5,2050),AVAL(5,2050)
        REAL SNRDB
        INTEGER BP,NFILT(5),ICODE,ANSWER,STORE(5)
        OPEN(UNIT=21,DEVICE='DSK')
        TYPE 900
900     FORMAT(4X,'OPERATOR TO NOTE :YOU ARE REQUIRED TO KEYIN VARIOUS
       1 DATA ,AS AND WHEN NOTIFIED ON THE TERMINAL. THIS IS DONE BY
       1 KEYINGIN THOSE DATA FOUND WITHIN " ",AS REQUIRED.')
        TYPE 901
901     FORMAT(4X,'PLEASE KEYIN NO. OF SUB-BANDS DESIRED IN THE CODER')
        ACCEPT*,K
C
C       K DENOTES NUMBER OF SUB-BANDS IN THE SUB-BAND CODER.
C
        TYPE 902
902     FORMAT(4X,'ENCODING TECHNIQUE ?')
        TYPE 903
903     FORMAT(4X,'KEYIN "1" IF APCM CODING DESIRED OR KEYIN "0"
       1 IF ADM CODING DESIRED')
        ACCEPT*,ICODE
        DO 10 BP=1,K
        IBP=BP
        CALL FILDES(IBP)
        TYPE 904
904     FORMAT(4X,'DO YOU WANT TO DISPLAY FILTER FREQUENCY PLOT?')
        TYPE 905
905     FORMAT(4X,'KEYIN "1" IF PLOT DESIRED,OTHERWISE KEYIN "0"')
        ACCEPT*,ANSWER
        IF(ANSWER.EQ.0)GO TO 100
        CALL GRAPH(IBP)
        TYPE 906
906     FORMAT(4X,'FREQUENCY PLOT IS STORED IN FILE FRQPLT.DAT')
100     CALL FLTIMP(IBP,1)
        CALL SRATE(IBP,0)
        STORE(IBP+1)=KK
        DO 20 I=1,KK
        HP3(IBP,I)=HP1(IBP,I)
20      CONTINUE
        CALL CODING(IBP,ICODE)
        TYPE 907
907     FORMAT(4X,'DO YOU WANT TO KNOW SUB-BAND SNR?')
```

```
      TYPE 908
 8    FORMAT(4X,'KEYIN "1" IF SNR DESIRED OTHERWISE KEYIN "0"')
      ACCEPT*,ANSWER
      IF(ANSWER.EQ.0)GO TO 200
      CALL SNR
      WRITE(5,909)SNRDB
 9    FORMAT(4X,'SUB-BAND SNR IS :',F12.6,' DB')
 0    CONTINUE
      DO 30 I=1,NN
      HP1(BP,I)=AVAL(BP,I)
      CONTINUE
      CALL SRATE1(IBP)
      CALL FLTIMP(IBP,0)
      CONTINUE
      CLOSE(UNIT=21,DEVICE='DSK')
      K1=2048;CALL SUM(K1,63)
      TYPE 910
10    FORMAT(4X,'DO YOU WANT TO KNOW OVERALL SNR?')
      TYPE 911
11    FORMAT(4X,'KEYIN "1" IF SNR DESIRED,OTHERWISE KEYIN "0"')
      ACCEPT*,ANSWER
      IF(ANSWER.EQ.0)GO TO 300
      DO 40 BP=1,K
      IBP=BP;KK=STORE(IBP+1)
      DO 50 I=1,KK
      HP1(IBP,I)=HP3(IBP,I)
0     CONTINUE
      CALL SRATE1(IBP)
      CALL FLTIMP(IBP,0)
0     CONTINUE
      K1=2048;CALL SUM(K1,60)
      CALL SNR
      CALL SEGSNR
      CALL PRISNR
00    CONTINUE
      TYPE 912
12    FORMAT(4X,'OPERATOR TO NOTE:THE PROGRAM EXECUTION IS OVER
    1 THE INPUT AND OUTPUT SPEECH SAMPLES ARE STORED IN FILES
    1 FILES FOR58.DAT AND FOR59.DAT RESPECTIVELY.')
      STOP
      END
```

```
      SUBROUTINE FILDES:-
           THIS SUBROUTINE DESIGNS FIR BANDPASS AND LOWPASS
           FILTERS FOR VARIOUS SUB-BANDS.
```

```
      SUBROUTINE FILDES(BP)
      INPUT:
      NFLT-FILTER LENGTH
      NBANDS-NUMBER OF BANDS
```

```
C       LGRID-GRID DENSITY ;SET TO 16
C       EDGE(2*NBANDS)-BAND EDGE ARRAY,LOWER AND UPPER EDGES
C                     FOR EACH BAND
C       FX(NBANDS)-DESIRED AMPLITUDE RESPONSE ARRAY.IT IS 1 IN
C       PASSBAND AND 0 IN STOPBAND.
C       WTX(NBAND)-WTD FUNCTION ARRAY IN EACH BAND.ERROR TO BE
C       WEIGHED BY 1 AND 10 RESPT.IN PASSBAND AND STOPBAND.
C       OUTPUT:
C          IMPULSE RESPONSE AND FREQUENCY REPONSE OF THE FILTER.
C

        COMMON /A1/DES,WT,ALPHA,IEXT,NFCNS,NGRID,PI2,AD,DEV,X,Y,GRID
        COMMON/A2/COUNT
        COMMON/A3/NX,NOLP,IBAND
        COMMON/A4/NFILT,HP
        DIMENSION IEXT(202),AD(202),ALPHA(202),X(202),Y(202)
        DIMENSION H(202),HPO(5,404),HP(5,12,404)
        DIMENSION DES(3232),GRID(3232),WT(3232)
        DIMENSION EDGE(20),FX(10),WTX(10),DEVIAT(10)
        DIMENSION OMEGA(50),RESPA(50)
        INTEGER BP,FIL1,FIL2,COUNT,NFILT(5),NX,NOLP,IBAND
        DOUBLE PRECISION OMEGA,SUMAR,SUMAC,RESPA,ATTN,RIPPLE,SVRIP,
     1  DELTAF
        DOUBLE PRECISION PI2,PI
        DOUBLE PRECISION AD,DEV,X,Y
        PI2=6.283185307179586
        PI=3.141592653589793
C
C       ARRAYS IEXT,AD,ALPHA,X,Y,H ARE DIMENSIONED NFMAX2+2
C       THE ARRAYS DES,GRID AND WT ARE DIMENSIONED 16(NFMAX/2+2)
C
        NFMAX=400
        FIL1=20
        IF(BP.EQ.0)FIL1=FIL1+1
        IF(BP.NE.0)COUNT=0
        READ(FIL1,*) NFILT(COUNT),NBANDS,LGRID
        IF (NFILT(COUNT).GT.NFMAX.OR.NFILT(COUNT).EQ.3)CALL ERROR
        IF(NBANDS.LE.0) NBANDS=1
C
C       GRID DENSITY IS ASSUMED TO BE 16 UNLESS SPECIFIED OTHERWISE
C
        IF(LGRID.LE.0) LGRID=16
        JB=2*NBANDS
        READ(FIL1,*) (FX(J),J=1,NBANDS)
        READ(FIL1,*) (WTX(J),J=1,NBANDS)
        IF(BP.EQ.0) GO TO 100
        DO 10 I=1,BP
        READ (FIL1,909)AAA
909     FORMAT(A1)
10      CONTINUE
100     READ(FIL1,*), (EDGE(J),J=1,JB)
        IF(NFILT(COUNT).EQ.0) GO TO 110
        NROX=0
        GO TO 120
```

```
110     READ(FIL1,*) RIPPLE,ATTN
        CLOSE(UNIT=20,DEVICE='DSK')
        RIPPLE=10**(RIPPLE/20)
        ATTN=10**(-ATTN/20)
        RIPPLE=(RIPPLE-1)/(RIPPLE+1)
        RIPPLE=DLOG10(RIPPLE)
        ATTN=DLOG10(ATTN)
        SVRIP=RIPPLE
        RIPPLE=(0.005309*(RIPPLE**2)+0.07114*RIPPLE-0.4761)*ATTN
     1  +(-0.00266*(RIPPLE**2)-0.5941*RIPPLE-0.4278)
        ATTN=11.01217+0.51244*SVRIP-0.51244*ATTN
        DELTAF=EDGE(3)-EDGE(2)
        NFILT(COUNT)=RIPPLE/DELTAF-ATTN*(DELTAF**2)+1
        IF(MOD(NFILT(COUNT),2).EQ.0)NFILT(COUNT)=NFILT(COUNT)+1
        NROX=1
120     NEG=0
        NODD=NFILT(COUNT)/2
        NODD=NFILT(COUNT)-2*NODD
        NFCNS=NFILT(COUNT)/2
        IF(NODD.EQ.1.AND.NEG.EQ.0) NFCNS=NFCNS+1
C
C       SET UP DENSE GRID.THE NUMBER OF POINTS IN GRID IS
C       (FILTER LENGTH+1)*GRID DENSITY/2
C
        GRID(1)=EDGE(1)
        DELF=LGRID*NFCNS
        DELF=0.5/DELF
        J=1
        L=1
        LBAND=1
140     FUP=EDGE(L+1)
145     TEMP=GRID(J)
C
C       CALCULATE THE MAGNITUDE RESPONSE AND THE WEIGHT
C       FUNCTION ON THE GRID.
C
        DES(J)=EFF(TEMP,FX,WTX,LBAND)
        WT(J)=WATE(TEMP,FX,WTX,LBAND)
        J=J+1
        GRID(J)=TEMP+DELF
        IF(GRID(J).GT.FUP) GO TO 150
        GO TO 145
150     GRID(J-1)=FUP
        DES(J-1)=EFF(FUP,FX,WTX,LBAND)
        WT(J-1)=WATE(FUP,FX,WTX,LBAND)
        LBAND=LBAND+1
        L=L+2
        IF(LBAND.GT.NBANDS) GO TO 160
        GRID(J)=EDGE(L)
        GO TO 140
160     NGRID=J-1
        IF(NEG.NE.NODD) GO TO 165
```

```
      IF(GRID(NGRID).GT.(0.5-DELF)) NGRID=NGRID-1
165   CONTINUE
C
C     SET UP A NEW APPROXIMATION PROBLEM WHICH IS EQUIVALENT
C     TO THE ORIGINAL PROBLEM.
C
      IF(NEG) 170,170,180
170   IF(NODD.EQ.1) GO TO 200
      DO 175 J=1,NGRID
      CHANGE=DCOS(PI*GRID(J))
      DES(J)=DES(J)/CHANGE
175   WT(J)=WT(J)*CHANGE
      GO TO 200
180   IF(NODD.EQ.1) GO TO 190
      DO 185 J=1,NGRID
      CHANGE=DSIN(PI*GRID(J))
      DES(J)=DES(J)/CHANGE
185   WT(J)=WT(J)*CHANGE
      GO TO 200
190   DO 195 J=1,NGRID
      CHANGE=DSIN(PI2*GRID(J))
      DES(J)=DES(J)/CHANGE
195   WT(J)=WT(J)*CHANGE
C
C     INITIAL GUESS FOR THE EXTERMAL FREQUENCIES--EQUALLY
C     SPACED ALONG THE GRID.
C
200   TEMP=FLOAT(NGRID-1)/FLOAT(NFCNS)
      DO 210 J=1,NFCNS
210   IEXT(J)=(J-1)*TEMP+1
      IEXT(NFCNS+1)=NGRID
      NM1=NFCNS-1
      NZ=NFCNS+1
C
C     CALL THE REMEZ EXCHANGE ALGORITHM TO DO THE APPROXIMATION
C     PROBLEM
C
      CALL REMEZ(EDGE,NBANDS)
C
C     CALCULATE THE IMPULSE RESPONSE.
C
      IF(BP.NE.0)COUNT=0
      IF(NEG) 300,300,320
300   IF(NODD.EQ.0) GO TO 310
      DO 305 J=1,NM1
305   H(J)=0.5*ALPHA(NZ-J)
      H(NFCNS)=ALPHA(1)
      GO TO 340
310   H(1)=0.25*ALPHA(NFCNS)
      DO 315 J=2,NM1
315   H(J)=0.25*(ALPHA(NZ-J)+ALPHA(NFCNS+2-J))
      H(NFCNS)=0.5*ALPHA(1)+0.25*ALPHA(2)
      GO TO 340
320   IF(NODD.EQ.0) GO TO 330
```

```
999     FORMAT(15X,F5.3,9X,F9.3)
        RETURN
        END
        FUNCTION EFF(TEMP,FX,WTX,LBAND)
C
C       FUNCTION TO CALCULATE THE DESIRED MAGNITUDE RESPONSE
C       AS A FUNCITON OF FREQUENCY.
C
        DIMENSION FX(5),WTX(5)
        EFF=FX(LBAND)
        RETURN
        END
        FUNCTION WATE(TEMP,FX,WTX,LBAND)
C
C       FUNCTION TO CALCULATE THE WEIGHT FUNCTION AS A FUNCTION
C                        OF FREQUENCY
C
        DIMENSION FX(5),WTX(5)
        WATE=WTX(LBAND)
        RETURN
        END
        SUBROUTINE ERROR
        TYPE 1
1       FORMAT('************ ERROR IN INPUT DATA *************')
        STOP
        END

C------------------------------------------------------------------
C
C       SUBROUTINE REMEZ(EDGE,NBANDS) :-
C       THIS SUBROUTINE IMPLEMENTS THE REMEZ EXCHANGE ALGORITHM
C       FOR THE WEIGHTED CHEBYCHEV APPROXIMATION OF A CONTINUOUS
C       FUNCTION WITH A SUM OF COSINES.INPUT TO THE SUBROUTINE
C       ARE A DENSE GRID WHICH REPLACES THE FREQUENCY AXIS,THE
C       DESIRED FUNCTION ON THIS GRID,THE WEIGHT FUNCTION ON THE
C       GRID,THE NUMBER OF COSINES AND AN INITIAL GUESS OF THE
C       EXTERNAL FREQUENCIES.THE PROGRAM MINIMIZES THE CHEBYCHEV
C       ERROR BY DETERMINING THE BEST LOCATION OF THE EXTREMAL
C       FREQUENCIES (POINTS OF MAXIMUM ERROR) AND THEN CALCULATES
C       THE COEFFICIENTS OF THE BEST APPROXIMATION.
C------------------------------------------------------------------

        SUBROUTINE REMEZ(EDGE,NBANDS)
        COMMON /A1/ DES,WT,ALPHA,IEXT,NFCNS,NGRID,PI2,AD,DEV,X,Y,GRID
        DIMENSION EDGE(20)
        DIMENSION IEXT(202),AD(202),ALPHA(202),X(202),Y(202)
        DIMENSION  DES(3232),GRID(3232),WT(3232)
        DIMENSION A(202),P(201),Q(201)
        DOUBLE PRECISION PI2,DNUM,DDEN,DTEMP,A,P,Q
        DOUBLE PRECISION AD,DEV,X,Y
C
C       THE PROGRAM ALLOWES A MAXIMUM NUMBER OF ITERATIONS OF 25
C
        ITRMAX=25
```

```
      DEVL=-1.0
      NZ=NFCNS+1
      NZZ=NFCNS+2
      NITER=0
100   CONTINUE
      IEXT(NZZ)=NGRID+1
      NITER=NITER+1
      IF(NITER.GT.ITRMAX) GO TO 400
      DO 110 J=1,NZ
      DTEMP=GRID(IEXT(J))
      DTEMP=DCOS(DTEMP*PI2)
110   X(J)=DTEMP
      JET=(NFCNS-1)/15+1
      DO 120 J=1,NZ
      JJJ=J
120   AD(J)=D(JJJ,NZ,JET)
      DNUM=0.0
      DDEN=0.0
      K=1
      DO 130 J=1,NZ
      L=IEXT(J)
      DTEMP=AD(J)*DES(L)
      DNUM=DNUM+DTEMP
      DTEMP=K*AD(J)/WT(L)
      DDEN=DDEN+DTEMP
130   K=-K
      DEV=DNUM/DDEN
      NU=1
      IF(DEV.GT.0.0) NU=-1
      DEV=-NU*DEV
      K=NU
      DO 140 J=1,NZ
      L=IEXT(J)
      DTEMP=K*DEV/WT(L)
      Y(J)=DES(L)+DTEMP
140   K=-K
      IF(DEV.GE.DEVL) GO TO 150
      CALL OUCH
      GO TO 400
150   DEVL=DEV
      JCHNGE=0
      K1=IEXT(1)
      KNZ=IEXT(NZ)
      KLOW=0
      NUT=-NU
      J=1
C
C        SEARCH FOR THE EXTREMAL FREQUENCIES OF THE BEST
C                          APPROXIMATION
C
200   IF(J.EQ.NZZ) YNZ=COMP
      IF(J.GE.NZZ) GO TO 300
      KUP=IEXT(J+1)
      L=IEXT(J)+1
```

```
      NUT=-NUT
      IF(J.EQ.2) Y1=COMP
      COMP=DEV
      IF(L.GE.KUP) GO TO 220
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 220
      COMP=NUT*ERR
210   L=L+1
      IF(L.GE.KUP) GO TO 215
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 215
      COMP=NUT*ERR
      GO TO 210
215   IEXT(J)=L-1
      J=J+1
      KLOW=L-1
      JCHNGE=JCHNGE+1
      GO TO 200
220   L=L-1
225   L=L-1
      IF(L.LE.KLOW) GO TO 250
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.GT.0.0) GO TO 230
      IF(JCHNGE.LE.0) GO TO 225
      GO TO 260
230   COMP=NUT*ERR
235   L=L-1
      IF(L.LE.KLOW) GO TO 240
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 240
      COMP=NUT*ERR
      GO TO 235
240   KLOW=IEXT(J)
      IEXT(J)=L+1
      J=J+1
      JCHNGE=JCHNGE+1
      GO TO 200
250   L=IEXT(J)+1
      IF(JCHNGE.GT.0) GO TO 215
255   L=L+1
      IF(L.GE.KUP) GO TO 260
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 255
      COMP=NUT*ERR
```

```
      GO TO 210
260   KLOW=IEXT(J)
      J=J+1
      GO TO 200
300   IF(J.GT.NZZ) GO TO 320
      IF(K1.GT.IEXT(1)) K1=IEXT(1)
      IF(KNZ.LT.IEXT(NZ)) KNZ=IEXT(NZ)
      NUT1=NUT
      NUT=-NU
      L=0
      KUP=K1
      COMP=YNZ*(1.00001)
      LUCK=1
310   L=L+1
      IF(L.GE.KUP) GO TO 315
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 310
      COMP=NUT*ERR
      J=NZZ
      GO TO 210
315   LUCK=6
      GO TO 325
320   IF(LUCK.GT.9) GO TO 350
      IF(COMP.GT.Y1) Y1=COMP
      K1=IEXT(NZZ)
325   L=NGRID+1
      KLOW=KNZ
      NUT=-NUT1
      COMP=Y1*(1.00001)
330   L=L-1
      IF(L.LE .KLOW) GO TO 340
      ERR=GEE(L,NZ)
      ERR=(ERR-DES(L))*WT(L)
      DTEMP=NUT*ERR-COMP
      IF(DTEMP.LE.0.0) GO TO 330
      J=NZZ
      COMP=NUT*ERR
      LUCK=LUCK+10
      GO TO 235
340   IF(LUCK.EQ.6) GO TO 370
      DO 345 J=1,NFCNS
345   IEXT(NZZ-J)=IEXT(NZ-J)
      IEXT(1)=K1
      GO TO 100
350   KN=IEXT(NZZ)
      DO 360 J=1,NFCNS
360   IEXT(J)=IEXT(J+1)
      IEXT(NZ)=KN
      GO TO 100
370   IF(JCHNGE.GT.0) GO TO 100
C
C     CALCULATION OF THE COEFFICIENTS OF THE BEST APPROXIMATION
```

```
C         USING THE INVERSE DISCRETE FOURIER TRANSFORM
C
400     CONTINUE
        NM1=NFCNS-1
        FSH=1.0E-06
        GTEMP=GRID(1)
        X(NZZ)=-2.0
        CN=2*NFCNS-1
        DELF=1.0/CN
        L=1
        KKK=0
        IF(EDGE(1).EQ.0.0.AND.EDGE(2*NBANDS).EQ.0.5) KKK=1
        IF(NFCNS.LE.3) KKK=1
        IF(KKK.EQ.1) GO TO 405
        DTEMP=DCOS(PI2*GRID(1))
        DNUM=DCOS(PI2*GRID(NGRID))
        AA=2.0/(DTEMP-DNUM)
        BB=-(DTEMP+DNUM)/(DTEMP-DNUM)
405     CONTINUE
        DO 430 J=1,NFCNS
        FT=(J-1)*DELF
        XT=DCOS(PI2*FT)
        IF(KKK.EQ.1) GO TO 410
        XT=(XT-BB)/AA
        FT=ACOS(XT)/PI2
410     XE=X(L)
        IF(XT.GT.XE) GO TO 420
        IF((XE-XT).LT.FSH) GO TO 415
        L=L+1
        GO TO 410
415     A(J)=Y(L)
        GO TO 425
420     IF((XT-XE).LT.FSH) GO TO 415
        GRID(1)=FT
        A(J)=GEE(1,NZ)
425     CONTINUE
        IF(L.GT.1) L=L-1
430     CONTINUE
        GRID(1)=GTEMP
        DDEN=PI2/CN
        DO 510 J=1,NFCNS
        DTEMP=0.0
        DNUM=(J-1)*DDEN
        IF(NM1.LT.1) GO TO 505
        DO 500 K=1,NM1
500     DTEMP=DTEMP+A(K+1)*DCOS(DNUM*K)
505     DTEMP=2.0*DTEMP+A(1)
510     ALPHA(J)=DTEMP
        DO 550 J=2,NFCNS
550     ALPHA(J)=2*ALPHA(J)/CN
        ALPHA(1)=ALPHA(1)/CN
        IF(KKK.EQ.1) GO TO 545
        P(1)=2.0*ALPHA(NFCNS)*BB+ALPHA(NM1)
        P(2)=2.0*AA*ALPHA(NFCNS)
```

```
      Q(1)=ALPHA(NFCNS-2)-ALPHA(NFCNS)
      DO 540 J=2,NM1
      IF(J.LT.NM1) GO TO 515
      AA=0.5*AA
      BB=0.5*BB
515   CONTINUE
      P(J+1)=0.0
      DO 520 K=1,J
      A(K)=P(K)
520   P(K)=2.0*BB*A(K)
      P(2)=P(2)+A(1)*2.0*AA
      JM1=J-1
      DO 525 K=1,JM1
525   P(K)=P(K)+Q(K)+AA*A(K+1)
      JP1=J+1
      DO 530 K=3,JP1
530   P(K)=P(K)+AA*A(K-1)
      IF(J.EQ.NM1) GO TO 540
      DO 535 K=1,J
535   Q(K)=-A(K)
      Q(1)=Q(1)+ALPHA(NFCNS-1-J)
540   CONTINUE
      DO 543 J=1,NFCNS
543   ALPHA(J)=P(J)
545   CONTINUE
      IF(NFCNS.GT.3) RETURN
      ALPHA(NFCNS+1)=0.0
      ALPHA(NFCNS+2)=0.0
      WRITE(5,1234)
      RETURN
      END

      DOUBLE PRECISION FUNCTION D(K,N,M)
C
C     FUNCTION TO CALCULATE THE LAGRANGE INTERPOLATION
C     COEFFICIENTS FOR USE IN THE FUNCTION GEE.
C
      COMMON /A1/ DES,WT,ALPHA,IEXT,NFCNS,NGRID,PI2,AD,DEV,X,Y,GRID
      DIMENSION IEXT(202),AD(202),ALPHA(202),X(202),Y(202)
      DIMENSION DES(3232),GRID(3232),WT(3232)
      DOUBLE PRECISION AD,DEV,X,Y
      DOUBLE PRECISION Q
      DOUBLE PRECISION PI2
      D=1.0
      Q=X(K)
      DO 3 L=1,M
      DO 2 J=L,N,M
      IF(J-K)1,2,1
1     D=2.0*D*(Q-X(J))
2     CONTINUE
3     CONTINUE
      D=1.0/D
      RETURN
```

```
      END
      DOUBLE PRECISION FUNCTION GEE(K,N)
C
C     FUNCTION TO EVALUATE THE FREQUENCY RESPONSE USING THE
C     LAGRANGE INTERPOLATION FORMULA IN THE BARYCENTRIC FORM
C
      COMMON /A1/ DES,WT,ALPHA,IEXT,NFCNS,NGRID,PI2,AD,DEV,X,Y,GRID
      DIMENSION IEXT(202),AD(202),ALPHA(202),X(202),Y(202)
      DIMENSION DES(3232),GRID(3232),WT(3232)
      DOUBLE PRECISION P,C,D,XF
      DOUBLE PRECISION PI2
      DOUBLE PRECISION AD,DEV,X,Y
      P=0.0
      XF=GRID(K)
      XF=DCOS(PI2*XF)
      D=0.0
      DO 1 J=1,N
      C=XF-X(J)
      C=AD(J)/C
      D=D+C
1     P=P+C*Y(J)
      GEE=P/D
      RETURN
      END
      SUBROUTINE OUCH
      TYPE 1
1     FORMAT('*********** FAILURE TO CONVERGE ***********'/
     1 '0PROBABLE CAUSE IS MACHINE ROUNDING ERROR'/
     2 '0THE IMPULSE RESPONSE MAY BE CORRECT'/
     3 '0CHECK WITH A FREQUENCY RESPONSE'/)
      RETURN
      END
C_____________________________________________________________________
C
C     SUBROUTINE GRAPH(BP):-
C          THIS SUBROUTINE PLOTS THE FREQUENCY- RESPONSE
C          OF THE FILTER DESIGNED BY SUBROUTINE FILDES(BP).
C     INPUT: FREQUENCY RESPONSE OF THE FILTER.
C     OUTPUT: FREQUENCY PLOT.
C
C_____________________________________________________________________
      SUBROUTINE GRAPH(BP)
      DIMENSION X(50),Y(50,1),A(158),IMAG4(5151)
      INTEGER BP,FIL2
      FIL2=50+BP
      OPEN(UNIT=6,DEVICE='DSK',FILE='FRQPLT.DAT')
      OPEN(UNIT=FIL2,DEVICE='DSK')
      OPEN(UNIT=25,DEVICE='DSK',FILE='FOR25.DAT')
      READ(25,*)A(145),A(146)
      READ(25,*)A(147),A(148)
      DO 10 I=1,50
      READ(FIL2,*)X(I),Y(I,1)
10    CONTINUE
      READ(25,999)(A(I),I=1,144)
```

```
999     FORMAT(72A1)
        A(149)=0.0
        CALL USPLX(X,Y,50,1,1,50,A,IMAG4,IER)
6       CONTINUE
        CLOSE(UNIT=6,DEVICE='DSK',FILE='FROPLT.DAT')
        CLOSE(UNIT=FIL2,DEVICE='DSK')
        CLOSE(UNIT=25,DEVICE='DSK')
        RETURN
        CALL UERTST
        CALL USMNMX
        END
C_____________________________________________________________________
C
C       SUBROUTINE FLTIMP(BP):-
C         THIS SUBROUTINE IMPLEMENTS THE FILTER DESIGNED BY
C         SUBROUTINE FILDES(BP)
C       INPUT: SPEECH SAMPLES AND FILTER IMPULSE RESPONSE.
C       OUTPUT:FILTERED SUB-BAND SIGNALS.
C
C_____________________________________________________________________


        SUBROUTINE FLTIMP(BP,MODE)
        COMMON/A2/COUNT
        COMMON/A3/NX,NOLP,IBAND
        COMMON/A4/NFILT,HP
        COMMON/A5/HP1,II
        EQUIVALENCE(N,II)
        DIMENSION X(2050),H(2050),Y(2050),IWK(100),HP(5,12,404)
        DIMENSION HP1(5,2050)
        COMPLEX CX(2050),CH(2050)
        INTEGER BP,MODE,NFILT(5),NX,NOLP,IBAND
        M=11;N=2**M
        DO 20 I=1,N
        X(I)=0.0
        H(I)=0.0
        COUNT=0;NX=BP
20      CONTINUE
        DO 30 I=1,NFILT(COUNT)
        H(I)=HP(COUNT+1,NX,I)
30      CONTINUE
        IF(MODE.EQ.0)GO TO 100
        OPEN(UNIT=47,DEVICE='DSK',FILE='SPEECH.DAT')
        READ(47,*)KL
        DO 40 I=1,KL
        READ(47,*)X(I)
40      CONTINUE
        GO TO 200
100     JJ=2049-NFILT(COUNT)
        DO 45 I=1,JJ
        X(I)=HP1(BP,I)
45      CONTINUE
200     DO 50 I=1,N
        CX(I)=CMPLX(X(I),0.)
```

```
      CH(I)=CMPLX(H(I),0.)
50    CONTINUE
      CALL FFT2(CX,M,IWK)
      CALL FFT2(CH,M,IWK)
      DO 60 I=1,N
      CX(I)=CX(I)*CH(I)
60    CONTINUE
      DO 70 I=1,N
      X1=REAL(CX(I));X2=-AIMAG(CX(I))
      CX(I)=CMPLX(X1,X2)
70    CONTINUE
      CALL FFRDR2(CX,M,IWK)
      CALL FFT2(CX,M,IWK)
      CALL FFRDR2(CX,M,IWK)
      DO 80 I=1,N
      Y(I)=REAL(CX(I))/FLOAT(N)
80    CONTINUE
      DO 90 I=1,N
      HP1(BP,I)=Y(I)
90    CONTINUE
      CLOSE(UNIT=47,FILE='SPEECH.DAT')
      RETURN
      END
C-----------------------------------------------------------------
C
C     SUBROUTINE SRATE(BP,OPTION):-
C      THIS SUBROUTINE IN CONJUCTION WITH SUBROUTINE SAMP(BP),
C      DECIMATES/INTERPOLATES THE SAMPLING RATES OF SUB-BAND
C      SIGNALS.
C
C-----------------------------------------------------------------
      SUBROUTINE SRATE(BP,OPTION)
      COMMON/A2/COUNT
      COMMON/A3/NX,NOLP,IBAND
      COMMON/A11/HINT
      INTEGER CHANEL,BP,COUNT,NX,NOLP,IBAND,OPTION,FIL3,HINT
      IF(OPTION.EQ.0) GO TO 100
      FIL3=22
      GO TO 200
100   FIL3=21
200   CONTINUE
      HINT=OPTION
      OPEN(UNIT=FIL3,DEVICE='DSK')
      REWIND FIL3
      COUNT=1
      IBAND=BP
      IF(BP.EQ.1)GO TO 300
      NOLP=6
      GO TO 400
300   NOLP=5
400   CONTINUE
600   READ (FIL3,*) CHANEL
      IF (CHANEL.EQ.BP) GO TO 500
```

```
999     FORMAT (/////,A1)
        GO TO 600
500     READ (FIL3,*) M,L
        NX=NOLP+IBAND
        CALL FILDES(0)
        CALL SAMP(BP,M,L)
        COUNT=COUNT+1
        NOLP=NOLP+1
        READ (FIL3,*,END=40) CHANEL
        IF (CHANEL.EQ.BP) GO TO 500
40      RETURN
        END
C_____________________________________________________________________
C
C       SUBROUTINE SAMP(BP,M,L):-
C         THIS SUBROUTINE DECIMATES/INTERPOLATES THE SAMPLING
C         RATES OF SUB-BAND SIGNALS .THIS SUBROUTINE CALLS SRINIT
C         TO INITIALIZE AND THEN CALLS SRCONV SUPPLYING I/P DATA
C         THROUGH BUFM AND TAKING OUTPUT DATA FROM BUFL.
C       INPUT:SUB-BAND SIGNAL AND IMPULSE REPONSE OF THE LP FILTER.
C       OUTPUT:DECIMATED/INTERPOLATED SUB-BAND SIGNALS.
C
C_____________________________________________________________________
        SUBROUTINE SAMP(BP,M,L)
        COMMON /SRCOM/ IQ,JQ,IL
        COMMON/A2/COUNT
        COMMON/A3/NX,NOLP,IBAND
        COMMON/A4/NFILT,HP
        COMMON/A5/HP1,NN
        COMMON/A11/HINT
        DIMENSION HP(5,12,404),HP1(5,2050)
        DIMENSION COEF(200),COFS(350),QBUF(650),ICTR(40)
        DIMENSION BUFL(3000),BUFM(3000)
        INTEGER BP,COUNT,NFILT(5),NX,NOLP,IBAND,HINT
        IQ=0
        JQ=0
        IL=0
        IF(HINT.EQ.0)GO TO 100
        DO 10 K=((NFILT(COUNT)-1)/2+1),NFILT(COUNT)
        COEF(K-(NFILT(COUNT)-1)/2)=HP(COUNT+1,NX,K)
10      CONTINUE
        GO TO 200
100     DO 20 K=((NFILT(COUNT)-1)/2+1),NFILT(COUNT)
        COEF(K-(NFILT(COUNT)-1)/2)=HP(COUNT+1,NX,K)*L
20      CONTINUE
200     CONTINUE
C
C       INITIALIZE CONVERSION ROUTINE.
C
        Q=(NFILT(COUNT)+L-1)/L
        NC=L*Q
        NI=2*L
        NQ=2*Q
        CALL SRINIT(M,L,QBUF,NQ,COEF,NFILT(COUNT),COFS,NC,ICTR,NI,IER
```

```
      TYPE *,IERR
      DO 30 J=1,NN
      BUFM(J)=HP1(BP,J)
30    CONTINUE
C
C     PROCESS DATA
C
      ND=(NN+M-1)/M
      CALL SRCONV(BUFM,BUFL,ND,QBUF,COFS,ICTR)
      NN=ND*L
      DO 40 I=1,NN
      HP1(BP,I)=BUFL(I)
40    CONTINUE
      RETURN
      END
C     ---------------------------------------------------------------
C
C     SUBROUTINE:SRINIT
C     INITIALIZATION FOR SRCONV WHICH CONVERTS THE SAMPLING
C     RATE OF A SIGNAL BY THE RATIO OF L/M
C
C     ---------------------------------------------------------------
      SUBROUTINE SRINIT(M,L,QBUF,NQ,COEF,N,COFS,NC,ICTR,NI,IERR )
      COMMON /SRCOM/ IQ,JQ,IL
      DIMENSION QBUF(1),COEF(1),COFS(1),ICTR(1)
C
C     M=DECIMATION RATIO
C     L=INTERPOLATION RATIO
C     QBUF=STATE VARIABLE BUFFER
C     NQ=SIZE OF QBUF,GREATER OR EQUAL TO 2*(THE NEXT GREATEST
C     INTEGER OF N/L)
C     COEF=ARRAY OF COEFFICIENTS FOR FIR INTERPOLATING FILTER
C     N=NO.OF TAPS IN FIR INTERPOLATING FILTER
C     COFS=SCRAMBLED COEFFICIENT VECTOR GENERATED BY SRINIT
C     NC=SIZE OF COFS VECTOR,EQUAL TO OR GREATER THAN
C     L*(THE NEXT GREATEST INTEGER OF N/L)
C     ICTR=CONTROL ARRAY GENERATED BY SRINIT AND USED BY SRCONV
C     NI=SIZE OF ICTR VECTOR EQUAL OR GREATER THAN 2*L
C     IERR=ERROR CODE FOR DEBUGGING
C         =0  NO ERROR FOUND IN INITIALIZATION
C         =1  QBUF(NQ) TOO SMALL
C         =2  COFS (NC) TOO SMALL
C         =3  ICTR (NI) TOO SMALL
C
      IERR=0
      IL=L
C
C     COMPUTE IQ
C
      IQ=N/L
      IF (N.NE.(IQ*L)) IQ=IQ+1
      NP=IQ*L
      IF (NQ.LT.(2*IQ)) IERR=1
```

```
      IF (NC.LT.NP) IERR=2
      NCF=(N+1)/2
      FL = L
C
C     ZERO OUT QBUF
C
      DO 10 I=1,NQ
      QBUF(I) = 0.0
10    CONTINUE
C
C     SCRAMBLE COEFFICIENTS
C
      I = 1
      DO 30 ML = 1,L
      DO 20 MQ = 1,IQ
      MX = (ML-1) + (MQ-1)*L
      IF (MX.LT.NCF) MM = NCF - MX
      IF (MX.GE.NCF) MM = MX-(N-NCF-1)
      IF (MM.LE.NCF) COFS(I) = COEF(MM)*FL
      IF (MM.GT.NCF) COFS(I) = 0.
      I = I+1
20    CONTINUE
30    CONTINUE
C
C     SET UP MOVING ADDRESS POINTER
C
      JQ = IQ
C
C     GENERATE CONTROL ARRAY ICTR
C
      LM = L*M
      IF (N1.LT.(2*L)) IERR = 3
      LC = 0
      MC = 0
      INCR = 0
      K = 1
      DO 50 I = 1,LM
      IF (LC.EQ.0) INCR = INCR+1
      IF (MC.LT.(M-1)) GO TO 40
C
C     NO OF SAMPLES TO UPDATE QBUF
C
      ICTR(K) = INCR
      INCR = 0
      K = K+1
C
C     STARTING LOCATION IN COFS VECTOR
C
      ICTR(K) = LC*IQ
      INCR = 0
      MC = -1
      K = K+1
40    LC = LC+1
      IF (LC.GE.L) LC = 0
```

```
      MC = MC+1
50    CONTINUE
      RETURN
      END
C
C-----------------------------------------------------------------------
C
C     SUBROUTINE : SRCONV
C       CONVERTS THE SAMPLING RATE OF A SIGNAL BY THE RATIO L/M.
C       SRINIT MUST BE CALLED PRIOR TO CALLING THIS ROUTINE.
C
C-----------------------------------------------------------------------
C
      SUBROUTINE SRCONV(BUFM,BUFL,ND,QBUF,COFS,ICTR )
      COMMON /SRCOM/ IQ,JQ,IL
      DIMENSION BUFM(1),BUFL(1),QBUF(1),COFS(1),ICTR(1)
C
C     BUFM = INPUT DATA BUFFER OF SIZE ND*M
C     BUFL = OUTPUT DATA BUFFER OF SIZE ND*L
C       ND = ANY POSITIVE INTEGER
C     QBUF = STATE VARIABLE BUFFER
C     COFS = SCRAMBLED COEFFICIENT VECTOR GENERATED BY SRINIT
C     ICTR = CONTROL ARRAY GENERATED BY SRINIT AND USED BY SRCONV
C
      MB = 1
      LB = 1
      L = IL
      DO 50  I = 1,ND
C
C     MB = INDEX ON BUFM
C     COMPUTE L OUTPUT SAMPLES
C
      K = 1
      DO 40  J = 1,L
      JD = ICTR(K)
      IC = ICTR(K+1)
       K = K+2
C
C     UPDATE QBUF
C
10    IF (JD.EQ.0) GO TO 20
      QBUF(JQ) = BUFM(MB)
      JQ1 = JQ + IQ
      QBUF(JQ1) = BUFM(MB)
      MB = MB+1
      JQ = JQ-1
      IF (JQ.EQ.0) JQ = IQ
      JD = JD-1
      GO TO 10
C
C     COMPUTE 1 SAMPLE OF OUTPUT DATA AND STORE IN BUFL
C
20    SUM = 0.
```

```
      DO 30 KQ = 1,10
      ICOF = KQ + IC
      IQB = KQ + JQ
      SUM = SUM + QBUF(IQB)*COFS(ICOF)
30    CONTINUE
      BUFL(LB) = SUM
      LB = LB + 1
40    CONTINUE
50    CONTINUE
      RETURN
      END

C-----------------------------------------------------------------
C
C     SUBROUTINE CODING:-
C          THIS SUBROUTINE CALLS THE SUBROUTINE FOR THE CODING
C          TECHNIQUE OPTED FOR IN THE MAIN PROGRAM.
C
C-----------------------------------------------------------------

      SUBROUTINE CODING(BP,ANSWER)
      REAL VALUIN,STSIZE
      INTEGER BP,ANSWER,CTOB
      IF(ANSWER.EQ.1) GO TO 100
      CALL SPATC(BP,1)
      CALL VARIAN(BP)
      CALL ADMCO(BP)
      GO TO 200
100   CALL VARIAN(BP)
      CALL APCMCO(BP)
      CALL APCMDE(BP)
200   CONTINUE
      RETURN
      END

C-----------------------------------------------------------------
C
C     SUBROUTINE VARIAN(BP):-
C        THIS SUBROUTINE CALCULATES MEAN,VARIANCE AND STANDARD
C        DEVIATION OF THE TIME SERIES OF SPEECH SAMPLES AT THE
C        INPUT TO THE ENCODER.
C     INPUT:DECIMATED SUB-BAND SIGNAL.
C     OUTPUT:SUB-BAND SIGNAL WITH UNIT VARIANCE.
C
C-----------------------------------------------------------------

      SUBROUTINE VARIAN(BP)
      COMMON/A5/HP1,K
      COMMON/A6/STDEV
      DIMENSION X(2050),HP1(5,2050)
      REAL SUM,AMEAN,VAR,S2,STDEV
      INTEGER BP
```

```
      OPEN(UNIT=60,DEVICE='DSK')
      OPEN(UNIT=61,DEVICE='DSK')
      WRITE(60,*)K
      SUM=0
      DO 10 I=1,K
      WRITE(60,*)BP1(BP,I)
      X(I)=BP1(BP,I)
      SUM=SUM+X(I)
10    CONTINUE
      AMEAN=SUM/K
      VAR=0
      DO 20 J=1,K
      S2=(X(J)-AMEAN)**2
      VAR=VAR+S2
20    CONTINUE
      VAR=VAR/(K-1)
      STDEV=SQRT(VAR)
      WRITE(61,*)K
      DO 30 I=1,K
      X(I)=X(I)/STDEV
      WRITE(61,*)X(I)
30    CONTINUE
      CLOSE(UNIT=60,DEVICE='DSK')
      CLOSE(UNIT=61,DEVICE='DSK')
      RETURN
      END
C---------------------------------------------------------------------
C
C     SUBROUTINE ADMCO(BP)
C          THIS SUBROUTINE PERFORMS ADM CODING AND DECODING OF
C          SUB-BAND SIGNALS.
C     INPUT:SUB-BAND SIGNAL.
C     OUTPUT:CODED AND DECODED SUB-BAND SIGNAL.
C
C---------------------------------------------------------------------
      SUBROUTINE ADMCO(BP)
      COMMON/A6/STDEV
      REAL SAMPLE,SS,SV(1:2050),STSIZE,DELMIN,DELMAX
      INTEGER IR(2050),IR1
      OPEN(UNIT=61,DEVICE='DSK')
      OPEN(UNIT=63,DEVICE='DSK')
      READ(61,*)N
      WRITE(63,*)N
      DO 10 I=1,N
      IR(I)=0
10    CONTINUE
      I=1
      SV(1)=0.0
      READ(61,*)SAMPLE
      IF(SAMPLE.GT.SV(1))GO TO 100
      IR1=-1
      GO TO 200
100   IR1=1
200   CONTINUE
```

```
      A=1.5
      B=0.6666
      DELMIN=STSIZE(BP);SS=DELMIN;DELMAX=100*DELMIN
      SV(I)=SV(1)+SS*IR1
300   I=I+1
      IF(SS.LT.DELMIN)SS=DELMIN
      IF(SS.GT.DELMAX)SS=DELMAX
      READ(61,*,END=20)SAMPLE
      IF(SAMPLE.GT.SV(I-1))GO TO 400
      IR(I)=-1
      GO TO 500
400   IR(I)=1
500   CONTINUE
      IF(IR(I).EQ.IR1)GO TO 600
      SS=-B*SS
      GO TO 700
600   SS=A*SS
700   IR1=IR(I)
      SV(I)=SV(I-1)+SS
      GO TO 300
20    CONTINUE
      DO 30 I=1,I
      SV(I)=SV(I)*STDEV
      WRITE(63,*)SV(I)
30    CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
C
C        SUBROUTINE APCMCO:-
C            THIS SUBROUTINE PERFORMS APCM ENCODING ON THE OUTPUT
C            OF THE DECIMATOR.
C        INPUT:SUB-BAND SIGNAL.
C        OUTPUT:CODED SUB-BAND SIGNAL.
C
C-----------------------------------------------------------------------
       SUBROUTINE APCMCO(BP)
       COMMON/A7/BIT,DELMIN
       REAL MULT(4,0:8),DELMIN,DELMAX,VALMIN
       INTEGER Y(4),BIT,BP,CTOB
       MULT(2,0)=0.85;MULT(2,1)=1.9
       MULT(3,0)=0.8455;MULT(3,1)=1.0;MULT(3,2)=1.0;MULT(3,3)=1.6
       MULT(4,0)=0.9;MULT(4,1)=0.9;MULT(4,2)=0.9;MULT(4,3)=0.9
       MULT(4,4)=1.2;MULT(4,5)=1.6;MULT(4,6)=2.0;MULT(4,7)=2.4
       BIT=CTOB(BP)
       DELMIN=VALMIN(BP)
       DELMAX=DELMIN*100
       S=DELMIN
       OPEN(UNIT=61,DEVICE='DSK')
       READ(61,*)N
       OPEN(UNIT=62,DEVICE='DSK')
       WRITE(62,*)N
       DO 10 I=1,N
       IF(S.LT.DELMIN) S=DELMIN
```

```
      IF(S.GT.DELMAX) S=DELMAX
      RANGE=S*2**(BIT-1)
      READ(61,*)SAMPLE
      SAMPLE=SAMPLE+RANGE
      DO 20 J=1,BIT
      Y(J)=0
      IF(SAMPLE.LT.RANGE) GO TO 20
      Y(J)=1.0
      SAMPLE=SAMPLE-RANGE
20    RANGE=RANGE/2
      WRITE(62,999)(Y(J),J=1,BIT)
999   FORMAT(1X,4I1)
      OUTPUT=0
      DO 40 J=2,BIT
      OUTPUT=2*OUTPUT+Y(J)
40    CONTINUE
      IF(Y(1).NE.1)OUTPUT=(2**(BIT-1))-OUTPUT-1
      S=S*MULT(BIT,OUTPUT)
10    CONTINUE
      CLOSE(UNIT=61,DEVICE='DSK')
      CLOSE(UNIT=62,DEVICE='DSK')
      RETURN
      END
C-----------------------------------------------------------------------
C
C     SUBROUTINE APCMDE(BP):-
C        THIS SUBROUTINE PERFORMS APCM DECODING .
C     INPUT:CODED SUB-BAND SIGNAL.
C     OUTPUT:DECODED SUB-BAND SIGNAL.
C
C-----------------------------------------------------------------------
      SUBROUTINE APCMDE(BP)
      COMMON/A6/STDEV
      COMMON/A7/BIT,DELMIN
      COMMON/A8/NN,AVAL
      DIMENSION AVAL(5,2050)
      REAL MULT(4,0:8),VALUE,MEAN,VAR,STDEV,DELMIN,DELMAX
      INTEGER Y(4),BIT,BP
      MULT(2,0)=0.85;MULT(2,1)=1.9
      MULT(3,0)=0.8455;MULT(3,1)=1.0;MULT(3,2)=1.0;MULT(3,3)=1.5
      MULT(4,0)=0.9;MULT(4,1)=0.9;MULT(4,2)=0.9;MULT(4,3)=0.9
      MULT(4,4)=1.2;MULT(4,5)=1.6;MULT(4,6)=2.0;MULT(4,7)=2.4
      DELMAX=DELMIN*160
      S=DELMIN
      OPEN(UNIT=62,DEVICE='DSK')
      OPEN(UNIT=63,DEVICE='DSK')
      READ(62,*)NN
      WRITE(63,*)NN
      DO 10 K=1,NN
      IF(S.LT.DELMIN) S=DELMIN
      IF(S.GT.DELMAX) S=DELMAX
      READ(62,999)(Y(J),J=1,BIT)
999   FORMAT(1X,4I1)
      IVALUE=0
```

```
      DO 20 I=2,BIT
      IVALUE=IVALUE*2+Y(I)
20    CONTINUE
      IF(Y(1).EQ.0)IVALUE=(2**(BIT-1))-IVALUE-1
      IVALUE=IVALUE+1
      VALUE=S*IVALUE-S/2
      IF(Y(1).EQ.0)VALUE=-VALUE
      VALUE=VALUE*STDEV
      WRITE(63,*)VALUE
      AVAL(BP,K)=VALUE
      OUTPUT=0
      DO 30 I=2,BIT
      OUTPUT=2*OUTPUT+Y(I)
30    CONTINUE
      IF(Y(1).NE.1)OUTPUT=(2**(BIT-1))-OUTPUT-1
      S=S*MULT(BIT,OUTPUT)
10    CONTINUE
      CLOSE(UNIT=62,DEVICE='DSK')
      CLOSE(UNIT=63,DEVICE='DSK')
      RETURN
      END
C______________________________________________________________________
C
C         FUNCTION CTOB(C):-
C               THIS FUNCTION PROVIDES THE NUMBER OF BITS USED FOR
C               ENCODING IN DIFFERENT BANDS.
C
C______________________________________________________________________
      FUNCTION CTOB(BP)
      INTEGER CTOB,BP
      IF(BP.EQ.1)CTOB=3
      IF(BP.EQ.2)CTOB=2
      IF(BP.EQ.3)CTOB=2
      IF(BP.EQ.4)CTOB=2
      IF(BP.EQ.5)CTOB=2
      RETURN
      END
C______________________________________________________________________
C
C         FUNCTION VALMIN(BP):-
C               THIS FUNCTION PROVIDES THE VALUES OF DELMIN FOR VARIOUS
C               SUB-BANDS.
C
C______________________________________________________________________
      FUNCTION VALMIN(BP)
      REAL VALMIN
      INTEGER BP
      IF(BP.EQ.1)VALMIN=0.01125
      IF(BP.EQ.2)VALMIN=0.013
      IF(BP.EQ.3)VALMIN=0.015
      IF(BP.EQ.4)VALMIN=0.0155
      IF(BP.EQ.5)VALMIN=0.018
      RETURN
      END
```

```
C------------------------------------------------------------------------
C
C        FUNCTION STSIZE:-
C              THIS FUNCTION PROVIDES THE INITIAL STEP SIZES FOR
C              VARIOUS SUB-BANDS IN ADP CODING.
C
C------------------------------------------------------------------------
         FUNCTION STSIZE(BP)
         REAL STSIZE
         INTEGER BP
         IF(BP.EQ.1)STSIZE=0.01930
         IF(BP.EQ.2)STSIZE=0.002
         IF(BP.EQ.3)STSIZE=0.0025
         IF(BP.EQ.4)STSIZE=0.003
         IF(BP.EQ.5)STSIZE=0.0035
         RETURN
         END

C------------------------------------------------------------------------
C
C        SUBROUTINE SRATE1(BP):-
C              THIS SUBROUTINE IN CONJUCTION WITH SUBROUTINE SAMP
C              INTERPOLATES THE SPEECH SAMPLES TO THE ORIGINAL
C              SAMPLING RATE.
C
C------------------------------------------------------------------------


         SUBROUTINE SRATE1(BP)
         COMMON/A2/COUNT
         COMMON/A3/NX,NOLP,IBAND
         COMMON/A11/HINT
         DIMENSION X(2050)
         INTEGER CHANEL,BP,COUNT,NX,NOLP,IBAND,HINT,FIL4
         IF(HINT.EQ.0)GO TO 100
         FIL4=24
         GO TO 110
 100     FIL4=23
 110     CONTINUE
         OPEN(UNIT=FIL4,DEVICE='DSK')
         IF(BP.EQ.1)GO TO 150
         COUNT=2
         GO TO 200
 150     COUNT=3
 200     CONTINUE
         IBAND=BP
         NOLP=7
         REWIND FIL4
 300     READ (FIL4,*) CHANEL
         IF (CHANEL.EQ.BP) GO TO 400
         READ (FIL4,*)JJ
         GO TO 300
 400     READ (FIL4,*) M,L
         COUNT=COUNT-1
```

```
      NOLP=NOLP-1
      NX=NOLP+ISAMP
      CALL SAMP(NP,N,M)
      READ (FILE,*,END=10) CHANEL
      IF (CHANEL.EQ.NP) GO TO 100
10    RETURN
      END
C------------------------------------------------------------------------
C
C     SUBROUTINE SUM:-
C        THIS SUBROUTINE SUMS UP THE OUTPUT OF ALL THE FILTERS
C     INPUT:SUB-BAND SIGNALS OF ALL THE SUB-BANDS.
C     OUTPUT:OUTPUT OF THE SUB-BAND CODER.
C
C------------------------------------------------------------------------
      SUBROUTINE SUM(K1,IFILNO)
      COMMON/A5/OP1,KKK
      DIMENSION OP1(5,2050),X(2050)
      INTEGER IFILNO,K1
      K1=2048
      DO 10 I=1,K1
      X(I)=0.0
      DO 20 J=1,4
      X(I)=X(I)+OP1(J,I)
20    CONTINUE
10    CONTINUE
      OPEN(UNIT=IFILNO,DEVICE='DSK')
      WRITE(IFILNO,*)K1
      DO 30 I=1,K1
      WRITE(IFILNO,*)X(I)
30    CONTINUE
      CLOSE(UNIT=IFILNO,DEVICE='DSK')
      RETURN
      END
C------------------------------------------------------------------------
C
C     SUBROUTINE SNR:-
C        THIS SUBROUTINE CALCULATES THE SIGNAL TO NOISE RATIO.
C     INPUT:INPUT AND OUTPUT OF THE SYSTEM.
C     OUTPUT:SIGNAL TO NOISE RATIO OF THE CODER IN DB.
C
C------------------------------------------------------------------------
      SUBROUTINE SNR
      COMMON/A9/SNRDB
      DIMENSION X(2050),Y(2050)
      REAL ANUMER,ADENMR,SNRDB
      ANUMER=0
      ADENMR=0
      OPEN(UNIT=60,DEVICE='DSK')
      OPEN(UNIT=63,DEVICE='DSK')
      READ(60,*)N
      READ(63,*)N
      DO 10 I=1,N
      READ(60,*)X(I)
```

```
      ANUMER=ANUMER+X(I)**2
      READ(63,*)Y(I)
      ADENMR=ADENMR+(Y(I)-X(I))**2
10    CONTINUE
      SNRDB=ALOG10(ANUMER/ADENMR)*10
      CLOSE(UNIT=60,DEVICE='DSK')
      CLOSE(UNIT=63,DEVICE='DSK')
      RETURN
      END
C______________________________________________________________________
C
C     SUBROUTINE SEGSNR:-
C        THIS SUBROUTINE CALCULATES SEGMENTAL SNR OF SUB-BAND
C        CODER.THE LENGTH OF THE SEGMENT IS 128 SAMPLES.
C     INPUT:INPUT AND OUTPUT SPEECH SAMPLES OF THE CODER.
C     OUTPUT: SEGMENTAL SIGNAL TO NOISE RATIO OF THE CODER.
C     (LENGTH OF THE SEGMENT=128 SPEECH SAMPLES)
C
C______________________________________________________________________
      SUBROUTINE SEGSNR
      COMMON/A10/SNRSEG
      DIMENSION X(2050),Y(2050)
      REAL ANUMER,ADENMR,SNRSEG,TH
      II=0
      SNRSEG=0.0
      TYPE 900
900   FORMAT(1X,'PLEASE SPECIFY THRESHOLD OF SNR(I)')
      TYPE 901
901   FORMAT(1X,'TH SPECIFIES THRESHOLD')
      ACCEPT*,TH
      OPEN(UNIT=60,DEVICE='DSK')
      OPEN(UNIT=63,DEVICE='DSK')
      READ(60,*)N
      READ(63,*)N
      DO 10 I=0,15
      READ(60,*)(X(J),J=128*I+1,128*I+128)
      READ(63,*)(Y(J),J=128*I+1,128*I+128)
      DO 20 J=128*I+1,128*I+128
      ANUMER=0.0
      ADENMR=0.0
      ANUMER=ANUMER+X(J)**2
      ADENMR=ADENMR+(X(J)-Y(J))**2
20    CONTINUE
      SNR=ALOG10(ANUMER/ADENMR)*10
      IF(SNR.GT.TH)GO TO 30
      SNR=0.0
      GO TO 40
30    SNRSEG=SNRSEG+SNR
      II=II+1
40    CONTINUE
10    CONTINUE
      SNRSEG=SNRSEG/II
      CLOSE(UNIT=60,DEVICE='DSK')
      CLOSE(UNIT=63,DEVICE='DSK')
```

```
      RETURN
      END
C---------------------------------------------------------------------
C
C     SUBROUTINE PRISNR:-
C      THIS SUBROUTINE PRINTS THE OVERALL AND SEGMENTAL SIGNAL
C      TO NOISE RATIO OF THE SUB-BAND CODER.
C
C---------------------------------------------------------------------
      SUBROUTINE PRISNR
      COMMON/A9/SNRDB
      COMMON/A10/SNRSEG
      REAL SNRDB,SNRSEG
      WRITE(5,995)
995   FORMAT(15X,'SNR PERFORMANCE'/
     1  15X,'==============='//)
      WRITE(5,996),SNRDB,SNRSEG
996   FORMAT(11X,'OVERALL SNR   =',F10.4,' dB'/
     1         11X,'SEGMENTAL SNR  =',F10.4,' dB'//
     2        11X,'*****************************'///)
      RETURN
      END
```